

Faculdade de Ciências da Universidade do Porto



# Árvores de Decisão Aplicadas à Detecção de Formas Costeiras Através de Imagens IKONOS-2

Dissertação

Mestrado em Engenharia Geográfica

Dário Miguel Ribeiro Ferreira

Julho 2013



Faculdade de Ciências da Universidade do Porto



## Árvores de Decisão Aplicadas à Detecção de Formas Costeiras Através de Imagens IKONOS-2

Trabalho realizado sob a orientação científica da Doutora Professora Ana Cláudia Teodoro, Professora Auxiliar do Departamento de Geociências, Ambiente e Ordenamento do Território da Faculdade de Ciências da Universidade do Porto e investigadora no Centro de Investigação em Ciências Geo-Espaciais, Faculdade de Ciências, Universidade do Porto, e sob co-orientação do Doutor Hernâni Gonçalves, Investigador no Centro de Investigação em Tecnologias e Sistemas de Informação em Saúde (CINTESIS), Faculdade de Medicina da Universidade do Porto

## **Agradecimentos**

Pela motivação, inspiração, carinho e tranquilidade agradeço à minha mulher Sara.

Pelo tempo dedicado, compreensão, ajuda e sentido crítico agradeço à orientadora Professora Ana Teodoro e ao co-orientador Hernâni Gonçalves.

## **Palavras-chave:**

Deteção remota, Algoritmos inteligentes, Árvores de decisão, Zonas costeiras, IKONOS-2.

## **Resumo:**

O processamento de imagens de satélite tem sido vastamente utilizado para análise e monitorização da superfície terrestre em muitas áreas e aplicações. Embora as zonas costeiras sejam as principais áreas habitáveis em Portugal, os estudos relativos à análise e evolução da costa e respetivas praias são escassos. O custo associado às técnicas tradicionais de aquisição de dados são um dos fatores que justificam esta situação. Face ao estado de arte em Portugal, o processamento de imagens de satélite de alta resolução assume um papel alternativo. Imagens IKONOS-2 foram usadas para a classificação de praias muito dinâmicas no noroeste de Portugal, entre a foz do rio Douro e a localidade da Aguda (concelho Vila Nova de Gaia), e para extração e quantificação da forma de corpos/formas costeiras, especificamente a restinga do rio Douro (Cabedelo), no período de 2001 a 2007. O objetivo deste trabalho prende-se com a avaliação do potencial das árvores de decisão para realizar tarefas de identificação e análise de corpos/formas costeiras. As árvores de decisão apresentam um bom desempenho na classificação de praias e na extração de formas corpos/formas costeiras, demonstrada pelos resultados obtidos na matriz de confusão, exatidão global e parâmetro estatístico kappa. Na classificação de praias, o algoritmo obteve uma exatidão global de 98.2% e um valor de kappa de 0.97. O algoritmo foi comparado com uma rede neuronal estabelecida em trabalhos anteriores para o mesmo conjunto de treino (Teodoro et al, 2010). As classificações de ambos os algoritmos são concordantes. Na extração da forma do Cabedelo, o algoritmo apresenta o valor mais baixo da exatidão global da classificação para o ano de 2001 com 96.8% e um valor de kappa de 0.89. Nos restantes anos, o algoritmo apresenta exatidões globais superiores a 98% e 0.95 para kappa. Este algoritmo foi comparado com o método GThE proposto em Teodoro e Gonçalves (2012). As formas extraídas de ambos os algoritmos são também concordantes. O Cabedelo apresenta a menor dimensão em 2001, com 173 040 m<sup>2</sup>, e a maior dimensão em 2007, com 265 744 m<sup>2</sup>.

**Keywords:**

Remote sensing, Data mining, Decision trees, Coastal areas, IKONOS-2.

**Abstract:**

Satellite images have been used to analyse and monitor the earth surface in many fields and applications. In Portugal, the coastal areas are zones with extreme importance for human activities since ever, however there are few studies about the evaluation and classification of beach features. The cost of making field surveys is very high due the amount of data necessary to make possible to analyse and predict the behavior of beach features. Thus, the satellite image processing plays a very important role, which can be considered as an alternative of traditional field surveys. IKONOS-2 images were used to classify and extract the form of coastal features in NW coast of Portugal, between the mouth of the river Douro and locality of Aguda. One image was used to classify beach features/patterns and another set of six images from 2001 to 2007 were used in the identification of a sand spit. The main goal of this work was the assessment of the potencial of decision trees algorithms to execute an identification and analysis of coastal features. The performance of the decision trees algorithm was evaluated by the confusion matrix, overall accuracy, and Kappa coefficient. In the classification of beach features/patterns, the algorithm presented an overall accuracy of 98.2% and a kappa coefficient of 0.97. The decision trees were compared with a neural network algorithm, and the results were in agreement. In the identification of the sand spit, the decision trees algorithm presented the lower performance was for the image of 2001, with an overall accuracy of 96.8% and 0.89 for kappa. For the remaining images (years), the overall accuracy presented values around 98% and 0.95 for kappa. These results were compared with the GThE method proposed in Teodoro e Gonçalves (2012). The shape of the sand spit identified with both methods was in agreement. The maximum area for the sand spit was obtained in the image of 2007, with 265 744 m<sup>2</sup>, and minimum value in image of 2001 with 173 040 m<sup>2</sup>.

## Índice

1. Introdução.....	1
1.1. Caraterização do ambiente costeiro .....	6
1.1.1. Área de estudo da classificação de praias.....	7
1.1.2. Área de estudo da extração de formas costeiras .....	9
1.2. Programa IKONOS.....	11
1.2.1. Conjunto de dados .....	13
2. Metodologia.....	14
2.1. Algoritmos Inteligentes.....	14
2.2. Árvores de Decisão em classificação de padrões .....	15
2.2.1. Critérios de Divisão .....	15
2.2.2. Critérios de Paragem .....	17
2.3. Poda da Árvore de Decisão.....	18
2.4. Avaliação da Precisão da Classificação .....	19
3. Resultados e Discussão.....	21
3.1. Classificação de Praias.....	21
3.2. Extração de formas costeiras .....	27
4. Conclusão .....	38
5. Bibliografia.....	39

## Índice de Figuras

Figura 1 Radiação electromagnética (adaptado de Fonseca e Fernandes, 2004) .....	2
Figura 2 Espectro electromagnético .....	3
Figura 3 Curvas de refletância espectral típicas da vegetação, solo e água (Davis e Swain, 1978) .....	4
Figura 4 Área de estudo da classificação de praias (GoogleEarth) .....	8
Figura 5 Área de estudo da extração de formas costeiras.....	10
Figura 6 Satélite IKONOS-2 .....	11
Figura 7 Imagem IKONOS-2 utilizada no trabalho .....	13
Figura 8 Imagem IKONOS-2. Banda NIR de dezembro de 2001 .....	13
Figura 9 Abordagens dos algoritmos inteligentes .....	14
Figura 10 Árvore de classificação de praias .....	21
Figura 11 Árvore de classificação podada para classificação de praias .....	22
Figura 12 Critério de paragem vs erro relativo .....	24
Figura 13 Imagem classificada .....	26
Figura 14 Imagem classificada com poda (cp=0.01 e xval=5).....	26
Figura 15 Áreas de treino da restinga. a) dezembro de 2001; b) junho de 2005 .....	27
Figura 16 Árvore de classificação para as imagens de junho de 2004 .....	28
Figura 17 Árvore de classificação para as imagens de junho 2005.....	29
Figura 18 Árvore de classificação para as imagens de junho de 2007 .....	29
Figura 19 Árvore de decisão podada para as imagens junho de 2004.....	30
Figura 20 Árvore de decisão podada para as imagens de junho de 2005 .....	30
Figura 21 Árvore de decisão podada para as imagens de junho de 2007 .....	30
Figura 22 Erro relativo em função do critério de paragem. a) junho de 2004; b) junho de 2005; c) junho de 2007 .....	31
Figura 23 20 Imagens classificados pelas árvores de classificação. a) dezembro de 2001; b) junho de 2004; c) julho de 2004; d) junho de 2005; e) setembro de 2005; f) junho de 2007.....	33
Figura 24 Forma do cabedelo em dezembro de 2001.....	34
Figura 25 Forma do cabedelo em junho de 2004 .....	34
Figura 26 Forma do cabedelo em julho de 2004 .....	35
Figura 27 Forma do cabedelo em junho de 2005 .....	35

Figura 28 Forma do cabedelo em setembro de 2005 .....	35
Figura 29 Forma do cabedelo em junho de 2007 .....	36
Figura 30 Ampliação da zona oeste do cabedelo, em junho de 2007.....	36

## **Índice de Tabelas**

Tabela 1 Informação orbital do satélite IKONOS .....	12
Tabela 2 Propriedades do sensor OSA .....	12
Tabela 3 Precisão para árvore de classificação de praias.....	22
Tabela 4 Precisão da árvore podada de classificação de praias .....	22
Tabela 5 Matriz de confusão para árvore de classificação de praias .....	24
Tabela 6 Matriz de confusão para.....	24
Tabela 7 Percentagem de cada classe na imagem .....	25
Tabela 8 Percentagem de cada classe na imagem .....	25
Tabela 9 Separabilidades das áreas de treino .....	27
Tabela 10 Precisão do classificador para as seis imagens .....	32
Tabela 11 Área do Cabedelo (m <sup>2</sup> ) (adaptado de Teodoro e Gonçalves, 2012) .....	37



## 1. Introdução

A deteção remota pode ser definida como a ciência e/ou arte de obter informação de um objeto, área ou fenómeno através da análise de dados adquiridos por sensores, instalados em satélites ou aeronaves, que não estão em contacto com o objeto, área ou fenómeno em estudo (Fonseca e Fernandes, 2004; Kiefer e Lillesand, 2000).

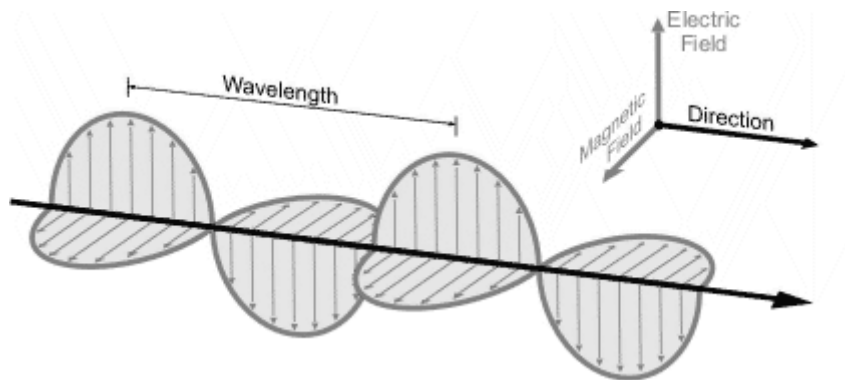
O processo de obtenção de informação do objeto em estudo baseia-se no facto de os objetos da superfície terrestre e oceânica refletirem radiação electromagnética (REM), sendo possível inferir as suas características físicas e químicas através da radiação electromagnética. As sucessivas interações da radiação com o meio de propagação, objetos de estudo e sensor proporcionam um envasamento dos dados disponibilizados para processamento de informação (Fonseca e Fernandes, 2004; Kiefer e Lillesand, 2000).

Os principais meios usados para a recolha de informação em detecção remota são os satélites, que embora tenham um elevado custo inicial, têm a capacidade de recolher imagens de uma determinada área da superfície, em diferentes datas segundo um mesmo ângulo de visualização, permitindo a deteção de alterações na superfície da área em causa. Não há estatísticas oficiais, mas estima-se que já foram lançados aproximadamente 4.600 satélites, e que apenas cerca de 500 continuam em funcionamento (NASA, 2013).

Os Estados Unidos da América através da NASA (National Aeronautics and Space Administration) e a Rússia através da Russian Federal Space Agency foram as primeiras potências mundiais a deterem programas espaciais, contudo a Europa através da ESA (European Space Agency), a China através da Chinese Space Program e o Japão através da JAXA (Japan Aerospace Exploration Agency) também entram na lista dos países a utilizar o espaço para fins militares, comerciais, de investigação e ajuda humanitária (Bally, 2005).

Como referido anteriormente, a deteção remota baseia-se no facto de os objectos da superfície reflectirem a REM proveniente do sol ou sensores ativos. A REM propaga-se através do espaço sem suporte material e apresenta um misto de propriedades ondulatórias e corpusculares (Fonseca e Fernandes, 2004). A radiação electromagnética é constituída por um campo eléctrico e um campo magnético, perpendiculares entre si (Figura 1).

Figura 1 Radiação electromagnética (adaptado de Fonseca e Fernandes, 2004)



Faraday mostrou que a variação no tempo de um campo magnético gera um campo eléctrico, e Maxwell mostrou que a variação no tempo de um campo eléctrico no tempo gera um campo magnético, existindo uma auto-sustentação da onda. Assim a direcção da propagação da onda é perpendicular aos campos eléctrico e magnético. Os fenómenos de interferência e difracção explicam-se através da teoria ondulatória, contudo o fenómeno fotoeléctrico explica-se com base na teoria corpuscular, proposta pelo físico Max Planck, em que estabelece que a REM é constituída por partículas designadas de fótons (Lillesand e Kiefer, 2000). Em detecção remota, analisam-se os fenómenos de propagação da REM à luz da teoria ondulatória e os fenómenos de interacção da radiação com a matéria à luz da teoria corpuscular (Fonseca e Fernandes, 2004).

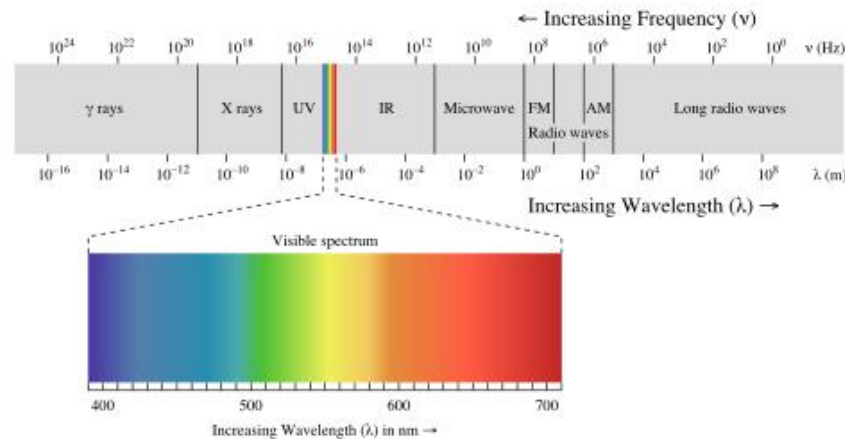
De acordo com a teoria de Planck, a quantidade de energia transportada por cada fóton numa onda electromagnética é dada por:

$$Q = \frac{h.c}{\lambda} \quad (1)$$

onde, Q: energia transportada por cada fóton; h: constante de Planck; c: velocidade da REM; e  $\lambda$ : comprimento de onda.

Verifica-se, assim, que a energia da radiação é inversamente proporcional ao comprimento de onda, ou seja, quanto maior o comprimento de onda, menor é a energia contida na radiação. Isto tem implicações importantes em detecção remota porque a radiação emitida em comprimentos de onda longos, como a microondas, tem mais dificuldade em ser detectada pelos sensores do que a radiação de pequenos comprimentos de onda, obrigando os sensores que funcionam com comprimentos de onda maiores (banda térmica do satélite Landsat) a receberem REM de um elemento do terreno de maiores dimensões (Fonseca e Fernandes, 2004; Lillesand e Kiefer, 2000).

Figura 2 Espectro electromagnético



A utilização de bandas e respetiva localização no espectro (Figura 2) está condicionada às janelas espectrais, que são zonas do espectro em que a atmosfera permite a passagem de REM proveniente do sol e da emissão da Terra, ou seja onde a absorção é baixa.

A radiação electromagnética incidente na matéria interage de três diferentes modos:

- *Absorção*: processo em que existe uma troca de energia entre os fótons e os átomos constituintes do corpo, resultando em transformações físicas deste;
- *Reflexão*: processo que consiste no reenvio de parte da radiação incidente para o ambiente envolvente, do mesmo lado da radiação incidente. Deste processo não resultam alterações do corpo nem das características da REM (ex.: vidro espelhado);
- *Transmissão*: a REM incidente desloca-se no corpo, sem interagir com as suas partículas, e é recuperada na face oposta à incidente (ex.: vidro simples).

Pelo princípio da conservação de energia temos que:

$$E_i(\lambda) = E_r(\lambda) + E_a(\lambda) + E_t(\lambda) \quad (2)$$

onde,  $E_i$ : energia incidente;  $E_r$ : energia reflectida;  $E_a$ : energia absorvida; e  $E_t$ : energia transmitida.

As proporções da energia refletida, absorvida e transmitida variam para diferentes objetos, que dependem das características físico-químicas dos materiais. Dentro que cada objeto, a refletância, absortância e transmitância variam nos diferentes comprimentos de onda (Lillesand e Kiefer, 2000). Essas diferenças permitem distinguir os objetos numa imagem.

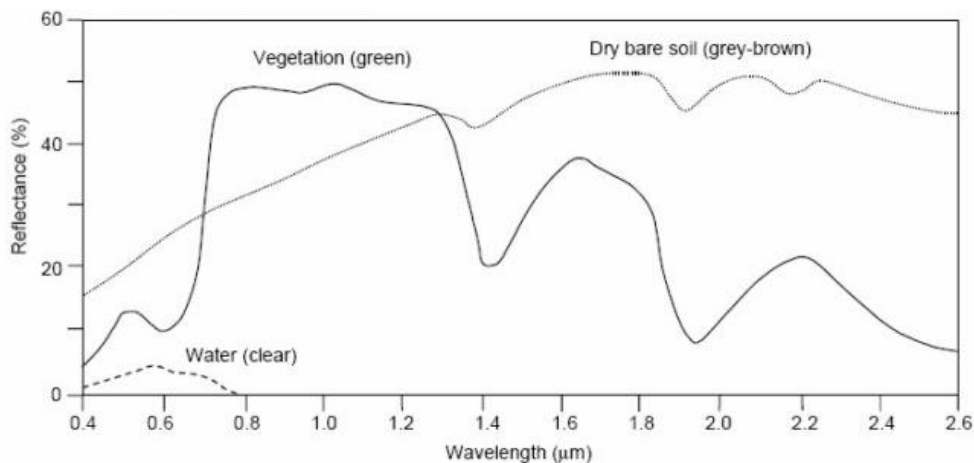
## Introdução

Como os sistemas de detecção remota operam principalmente nos comprimentos de onda onde a reflectância é predominante, é conveniente reescrever a equação (2) da seguinte forma:

$$E_r(\lambda) = E_i(\lambda) - [E_a(\lambda) + E_t(\lambda)] \quad (3)$$

Assim, quanto maior for a reflexão, menor a absorção e a transmissão.

Figura 3 Curvas de refletância espectral típicas da vegetação, solo e água (Davis e Swain, 1978)



Os vales ilustrados na Figura 3 e observáveis nos comprimentos de onda 0.45 e 0.67 μm, na curva de refletância espectral da vegetação, são devidos à presença de pigmentos nas folhas das plantas, como a clorofila, que absorvem fortemente a energia nesses comprimentos de onda. Quanto aos valores visíveis nos comprimentos de onda 1.4, 1.9, e 2.7 μm são resultantes da presença de água na constituição das folhas, que absorve especificamente nesses comprimentos de onda. O aumento da refletância no intervalo 0.7 a 1.3 μm é devido à estrutura das folhas da própria planta.

A curva de refletância do solo é mais linear do que a da vegetação, pois os fatores, como o teor de humidade, textura do solo, rugosidade da superfície, presença de óxidos de ferro e teor de matéria orgânica, que influenciam a reflexão da REM não são tão específicos como os supra-referidos.

Os corpos de água “pura” absorvem pouca radiação com comprimentos de onda inferiores a 0.6 μm, e a elevada transmitância (reduzida refletância) tem um máximo na zona azul-verde do espectro. Contudo a introdução de sedimentos sólidos suspensos, orgânicos ou inorgânicos na água aumenta a sua refletância, assim como a presença de algas (Lillesand e Kiefer, 2000). Estas alterações são usadas para monitorizar/detectar poluição ou eutrofização. Os corpos de água “pura” são facilmente identificados na zona do infravermelho próximo, pois não reflectem REM neste comprimento de onda, ao invés dos restantes objectos, como é possível verificar na curva de refletância espectral da água na Figura 3.

As atividades realizadas durante esta dissertação encontram-se apresentadas num enquadramento teórico, seguido de duas secções, relativas à seguinte ordem de trabalhos:

- i. Secção I: Classificação de praias;
- ii. Secção II: Determinação e quantificação de formas;

A secção I descreve o processo de classificação da praia da zona costeira em estudo em cinco classes, baseado em trabalhos anteriores. Os resultados da classificação com árvores de decisão podadas e não podadas são avaliados e comparados com os resultados de uma rede neuronal implementada em Teodoro *et al.*, (2011) para o mesmo conjunto de treino

A secção II apresenta uma metodologia de extração da área do Cabedelo (Foz do rio Douro) em seis épocas diferentes. Os resultados são comparados com dados de referência e com os resultados obtidos por técnicas clássicas e pelo método GThE (Teodoro and Gonçalves, 2012).

Em cada secção referência-se a metodologia utilizada, os resultados obtidos e a devida análise dos mesmos, considerando ainda a comparação com trabalhos anteriores. A comparação permite a comparação do desempenho das árvores de decisão com outros métodos previamente testados.

Em todos os procedimentos, foram utilizadas árvores de decisão, implementadas no *software open source* (R 2.15.3) e com pré e pós-processamento no *software* Matlab®.

### 1.1. Caraterização do ambiente costeiro

As áreas costeiras são zonas de extrema importância por se verificarem fortes interações entre a biosfera, atmosfera e hidrosfera. Estas interações têm uma relevância tão significativa que dois terços da população mundial habita a menos de 100 km da costa.

Em Portugal, inúmeras notícias são transmitidas sobre a erosão das praias e da alteração da morfologia das mesmas. A erosão da costa litoral é uma ação natural e que ocorre desde sempre pela natureza, mas em Teodoro *et al.* (2007) várias razões são apresentadas para o aumento dos eventos de erosão na zona norte da costa portuguesa, tais como a redução do fluxo de sedimentos consequente da construção das barragens no rio Douro, as atividades de dragagem para a indústria dos minerais, a construção de estruturas de defesa costeira como esporões e enrocamentos, e ocupação antrópica.

A maioria dos estudos da costa litoral portuguesa focam-se principalmente na praia emersa, na avaliação do perfil da praia e sua evolução, no recuo da linha de costa e galgamentos de eventuais dunas. Contudo existem poucos estudos que analisem a evolução da praia submersa e a respetiva análise da transferência de sedimentos entre a praia emersa e submersa, assim como a transferência de sedimentos pela corrente de deriva litoral numa escala temporal alargada, impossibilitando a modelação da morfologia das praias com rigor.

A dinâmica observada numa praia entre todos os intervenientes tem um cariz complexo, e a avaliação dos comportamentos e classificação da hidromorfologia de uma praia não é uma tarefa simples. Esta dinâmica complexa resulta da não-linearidade das interações entre as componentes aquáticas (marés e ondas), fronteiras sólidas naturais (praias e dunas), as fronteiras sólidas artificiais (esporões e enrocamentos) e os ventos, agravadas pela dificuldade em obter dados de monitorização (hidrodinâmicos, morfológicos e topográficos) contínuos no tempo e principalmente fiáveis (Teodoro *et al.*, 2009).

O setor de turismo é responsável pela geração de aproximadamente 10% do produto interno bruto nacional, e as principais zonas de estabelecimento do setor encontram-se nas áreas costeiras. Assim, a compreensão destas interações não se deve limitar às componentes visíveis na praia emersa, e deve aprofundar-se o conhecimento na interação de todas as variáveis, salientando as formas e padrões morfológicos e hidrodinâmicos denominados de hidroformas e hidromorfologias que revelam parte da interação da praia submersa e emersa. A capacidade de entender esta alteração espacial e temporal permite definir orientações politico-estratégicas para melhorar a eficácia da gestão costeira e criar sinergias com o setor da economia.

Embora as praias sejam os locais onde se processam mais as interações entre o meio marinho e terrestre devido à extensão e largura (em algumas praias), existem outras zonas de menor dimensão onde ocorrem as interações acima definidas e que apresentam uma importância portuária e biológica extrema. Estes locais são designados por restingas e situam-se na interface entre um estuário e o mar.

## Introdução

Uma restinga consiste numa acumulação de sedimentos arenosos provenientes da erosão fluvial ao longo do seu curso de água. A sucessiva acumulação de material arenoso fluvial levaria a uma rápida expansão do tamanho da restinga, contudo o material não é consolidado e facilmente é deslocado devido à dinâmica fluvial e marinha verificada no local. A forma e dimensão da restinga é variável no tempo devido às dinâmicas anteriormente referidas, e a sua evolução pode ser condicionada pela introdução de estruturas artificiais, como por exemplo esporões. Por definição (Bird, 2008), a restinga numa das extremidades está conectada a terra e apresenta a outra face exposta em águas abertas.

Várias metodologias para a classificação morfológica de praias e formas costeiras foram propostas por diferentes autores (Wright and Short, 1984; Sunamura, 1998) baseadas nas condições de maré, ondulação/agitação marítima e parâmetros de sedimentos. Em Portugal, estes argumentos necessários para o modelo de classificação das praias encontram-se inacessíveis, desatualizados, em quantidade insuficiente, ou não existem para alguns locais (Pais-Barbosa, 2007). Assim, a utilização dos modelos acima referidos requer a medição in-situ destes parâmetros, em escalas espaciais e temporais por vezes longas. A medição in-situ contempla um custo elevado devido à quantidade das amostragens e ao tempo de amostragem, aos equipamentos necessários para a medição, e às deslocações até ao local de estudo. Em casos extremos, a acessibilidade ao local de estudo poderá ser difícil e morosa aumentando o custo associado.

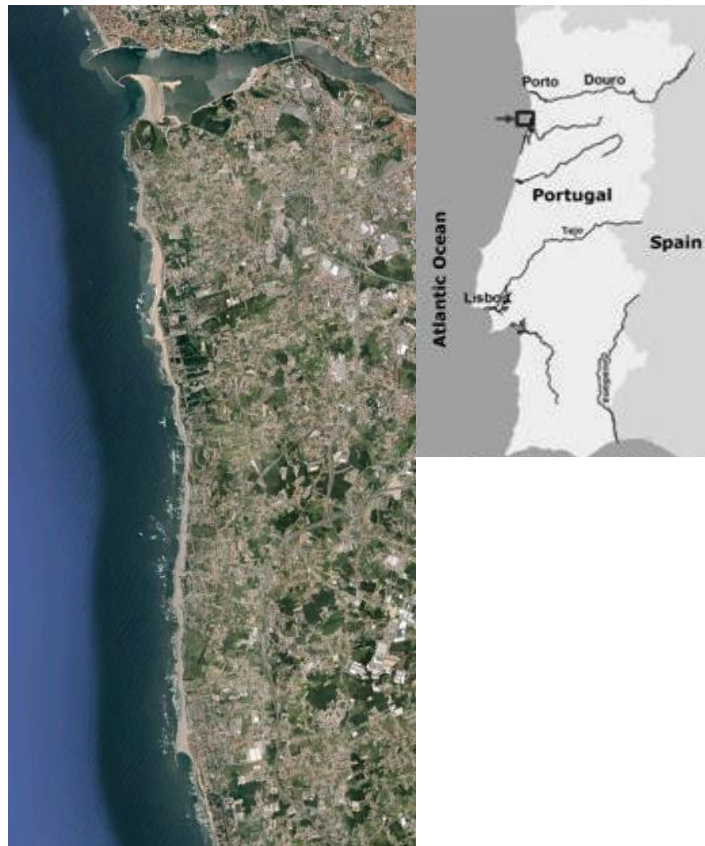
Face às limitações de dados em Portugal e às desvantagens associadas à obtenção dos dados para os referidos modelos de classificação, a análise e classificação da morfologia das praias através de imagens de satélite de alta resolução apresenta-se como uma alternativa ao nível de precisão, análise de dados de longa escala temporal, e económica.

No presente trabalho apresenta-se uma metodologia de classificação de formas costeiras através da utilização de imagens de satélite IKONOS-2.

### **1.1.1. Área de estudo da classificação de praias**

Neste trabalho apresenta-se uma metodologia para a classificação de praias baseada em árvores de decisão. Com intuito de tornar esta metodologia comparável com metodologias apresentadas em estudos anteriores (Teodoro *et al.*, 2009; Teodoro *et al.*, 2010; Teodoro *et al.*, 2011), selecionou-se a mesma área de estudo. Esta área situa-se na costa noroeste de Portugal (concelho de Vila Nova de Gaia), limitada a norte pela foz do rio Douro e a sul pela localidade da Aguda (Figura 4).

Figura 4 Área de estudo da classificação de praias (GoogleEarth)



Com uma extensão (norte-sul) aproximada de 9.5 km, esta área caracteriza-se pela frágil e muito dinâmica área costeira que encontra-se constantemente em alterações.

A área de estudo apresenta uma zona de praia muito dinâmica, facilmente ajustável às condições de agitação marítima e de marés. A movimentação de sedimentos pode ser tão poderosa que o perfil da praia e os padrões hidrodinâmicos numa escala temporal de dias podem assumir outras formas, melhores adaptadas às condições marinhas.

A maré apresenta um regime semi-diurno, podendo existir uma variação de 4 metros na época de primavera, sendo a direção norte-sul a direção principal da deriva litoral (Teodoro *et al.*, 2009).

A ondulação é caracterizada por ondas com uma altura média entre os 2 e os 3 metros, com um intervalo de ocorrência de 8 a 12 segundos. Em períodos de tempestade, as ondas podem exceder os 8 metros de altura com intervalos de ocorrência entre 16 a 18 segundos. A ondulação apresenta a maior frequência e intensidade no quadrante noroeste com 43.8% das ocorrências, na direção oés-noroeste (WNW) com 28.0% e nor-noroeste (NNW) com 21.2% (Velo-Gomes *et al.*, 2006).

Considerando a estrutura geológica, o local de estudo é uma área arenosa plana, constituída por praias e dunas arenosas com pouca elevação, com dinâmicas



hidroformas e hidromorfológicas costeiras que alteram a sua forma com muita frequência.

Na região sul da área de estudo alguns casos de cheias foram verificados. As causas deste incidentes ambientais foram identificados como a resposta da morfologia da praia à construção do esporão na Aguda, e da diminuição do transporte de sedimentos pelo rio Douro (Teodoro *et al.*, 2007).

### Áreas de treino

Por forma a tornar possível a analogia entre metodologias de classificação, as áreas de treino utilizadas são iguais às áreas de treino usadas nos trabalhos de Teodoro *et al.* (2009), Teodoro *et al.* (2010) e Teodoro *et al.* (2011). Deste modo foram utilizadas cinco classes: Mar, Sedimentos em Suspensão (SS), Zona de Rebentação (ZR), Areia Húmida (AH), Areia Seca (AS).

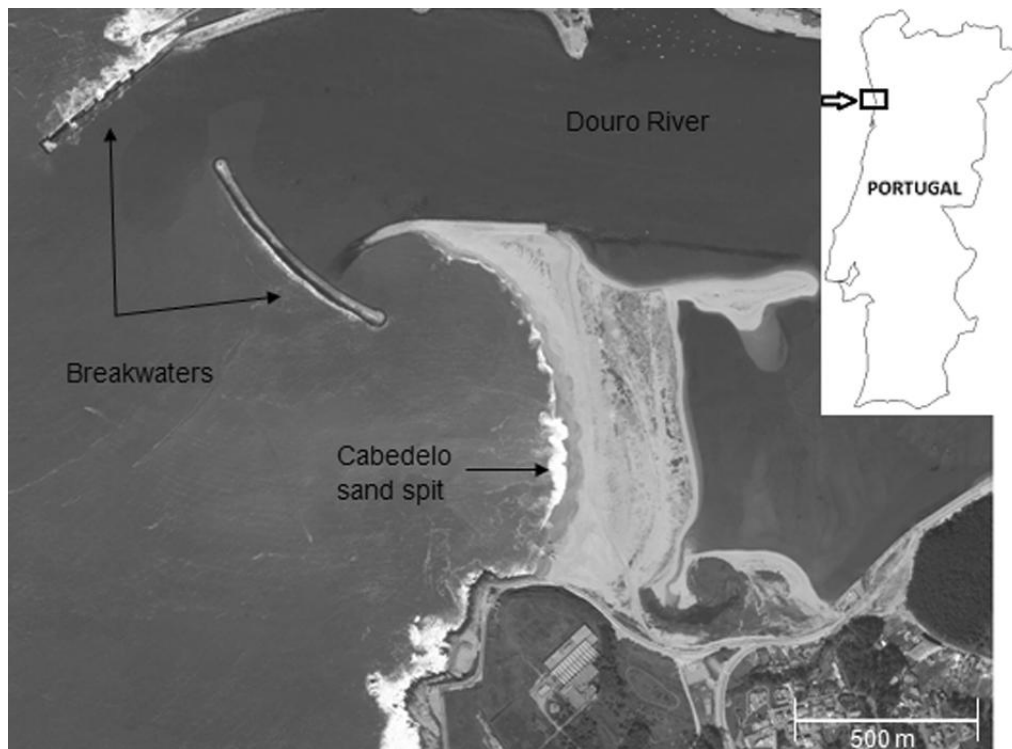
A definição, avaliação e justificação das áreas de treino usadas podem ser consultadas na bibliografia referida.

#### **1.1.2. Área de estudo da extração de formas costeiras**

Na segunda abordagem da aplicação das árvores de decisão, a metodologia proposta é aplicada à extração e quantificação da restinga situada na foz do rio Douro (Figura 5).

O estuário do Douro tem uma extensão de cerca de 20 hectares, englobando a Baía de São Paio (frente do estuário) e a zona arenosa do Cabedelo (parte final do estuário). O Cabedelo é uma área de grande beleza natural integrada na Rede Nacional de Áreas Protegidas com cerca de 800 m de extensão e 300 m de largura média.

Figura 5 Área de estudo da extração de formas costeiras(cabedelo)



A origem da restinga proporciona-se devido ao transporte de sedimentos do rio Douro, que na sua fase final transporta materiais finos e arenosos (processo de erosão dos sedimentos na cadeia de transporte). Os sedimentos acumulam-se junto à foz por difração da agitação marinha, que funciona como uma barreira de proteção contra a ação das ondas, especialmente nos casos de tempestade.

O rio Douro apresenta um caudal médio anual num ano típico de  $450 \text{ m}^3 \text{ s}^{-1}$  ( $700 \text{ m}^3 \text{ s}^{-1}$  e  $200 \text{ m}^3 \text{ s}^{-1}$  em anos húmidos e secos, respetivamente) na barragem de Crestuma. Este caudal apresenta uma elevada variabilidade entre os meses de inverno e os de verão, podendo em situações de cheia exceder os  $10\,000 \text{ m}^3 \text{ s}^{-1}$  (INAG, 2000). Esta variabilidade de caudal é uma das causas da modificação constante da forma e tamanho da restinga.

Em acréscimo, entre abril de 2004 e março de 2007, duas estruturas antropogénicas foram construídas na foz do rio. A norte foi construído um molhe com 350 metros e uma extensão de enraizamento de 200 m para terra e 7 metros de altura acima do nível das águas do mar. A sul foi implementado uma quebramar destacado, parcialmente submerso, com 450 metros de extensão e 6 metros de altura.

Estas estruturas, condicionam a dinâmica marinha e fluvial e consequentemente o transporte de sedimentos, que ocorreriam de forma natural. Embora as estruturas condicionem a dinâmica, a restinga continua a sofrer modificações constantes na sua forma e tamanho.

## Introdução

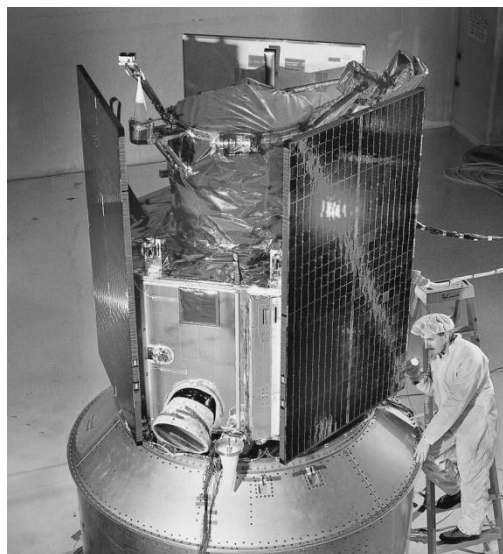
Como referido anteriormente, a restinga apresenta duas extremidades uma unida a terra e outra exposta às águas abertas. A extremidade exposta tem sofrido uma deslocação para o interior do estuário, sendo importante uma monitorização continua deste corpo arenoso.

Os principais trabalhos de monitorização da área em estudo tem sido realizados por técnicas de posicionamento GPS em modo diferencial dGPS (Baptista *et al.*, 2008), com uma periodicidade trimestral. Neste trabalho apresenta-se uma metodologia alternativa aos trabalhos de campo para a monitorização da restringa.

## 1.2. Programa IKONOS

O sistema satélite IKONOS foi concebido pela instituição Lockheed Martin Corporation como um sistema de deteção remota comercial. Dois satélites foram inicialmente previstos: o IKONOS-1, mas uma falha aquando do lançamento impediu o satélite de entrar em órbita, e o IKONOS-2 lançado em 1999 com sucesso, ainda permanece ativo atualmente (Figura 6).

Figura 6 Satélite IKONOS-2



O satélite apresenta uma órbita héliossíncrona quase polar com um período de revolução de 98 minutos. O satélite encontra-se a uma altitude de 681-709 km e dispõe de uma largura de varrimento entre 13 a 70 km (Tabela 1).

O trajeto de varrimento repete-se a cada 11 dias, contudo a resolução temporal das imagens é inferior, dada a área de sobreposição das imagens.

**Tabela 1 Informação orbital do satélite IKONOS**

<b>Operador</b>	GeoEye
<b>Data Lançamento</b>	24 de setembro de 1999
<b>Altitude</b>	681-709 km
<b>Orbita</b>	Héliossíncrona quase polar
<b>Período de revolução</b>	98 minutos
<b>Largura de varrimento</b>	13-70 km
<b>Inclinação</b>	98.1 graus
<b>Nodo descendente</b>	10h30 (AM)
<b>Resolução temporal</b>	11 dias
<b>Área da imagem</b>	11 km x 11 km
<b>Velocidade</b>	7 km/s
<b>Sensor</b>	OSA - <i>Optical Sensor Assembly</i>

O satélite tem a bordo o sensor OSA de muito alta resolução, que tem a capacidade de obter as imagens em dois modos: pancromático com resolução espacial de 1 metro, e multiespectral relativo às bandas da região do visível e do infra-vermelho próximo com resolução espacial de 4 metros (Tabela 2).

**Tabela 2 Propriedades do sensor OSA**

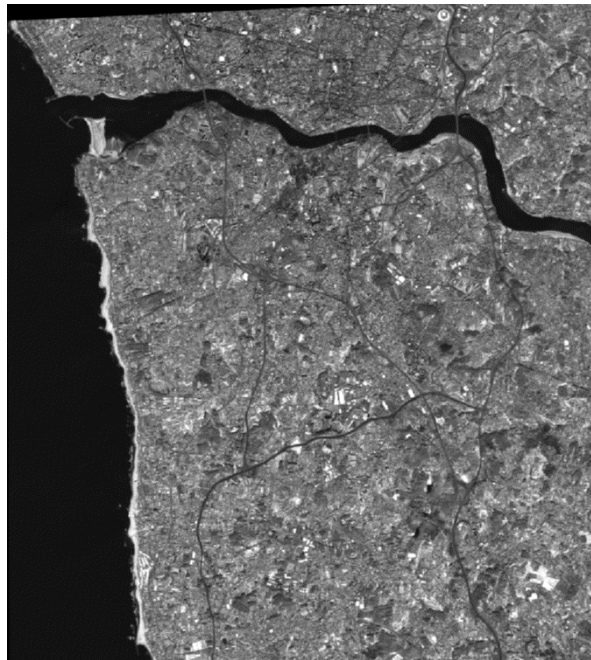
<b>Modo</b>	<b>Banda</b>	<b>Banda espectral</b>	<b>Resolução espacial</b>
Pancromático	-	0.49 - 0.90 $\mu\text{m}$	1 x 1 m
Multiespectral	1	0.45 - 0.52 $\mu\text{m}$	4 x 4 m
	2	0.52 - 0.60 $\mu\text{m}$	4 x 4 m
	3	0.63 - 0.69 $\mu\text{m}$	4 x 4 m
	4	0.76 - 0.90 $\mu\text{m}$	4 x 4 m

### 1.2.1. Conjunto de dados

No presente trabalho foram utilizadas 7 imagens IKONOS-2. Cada imagem é composta por 4 bandas (zona do visível – RGB, e infravermelho próximo – NIR).

Na secção I foram utilizadas as 4 bandas da imagem ilustrada na Figura 7, fornecida pela Agencia Espacial Europeia (ESA – *European Spatial Agency*)

Figura 7 Imagem IKONOS-2 utilizada no trabalho



Na secção II, foram utilizadas 6 imagens IKONOS-2 obtidas em seis períodos temporais diferentes. Para cada imagem foram utilizadas as 4 bandas relativas ao visível e infravermelho. Na Figura 8, é apresentada a banda do infravermelho próximo (NIR), do dia 24 de dezembro de 2011, já delimitada à area de estudo (restinga no Douro).

Figura 8 Imagem IKONOS-2. Banda NIR de dezembro de 2001



## 2. Metodologia

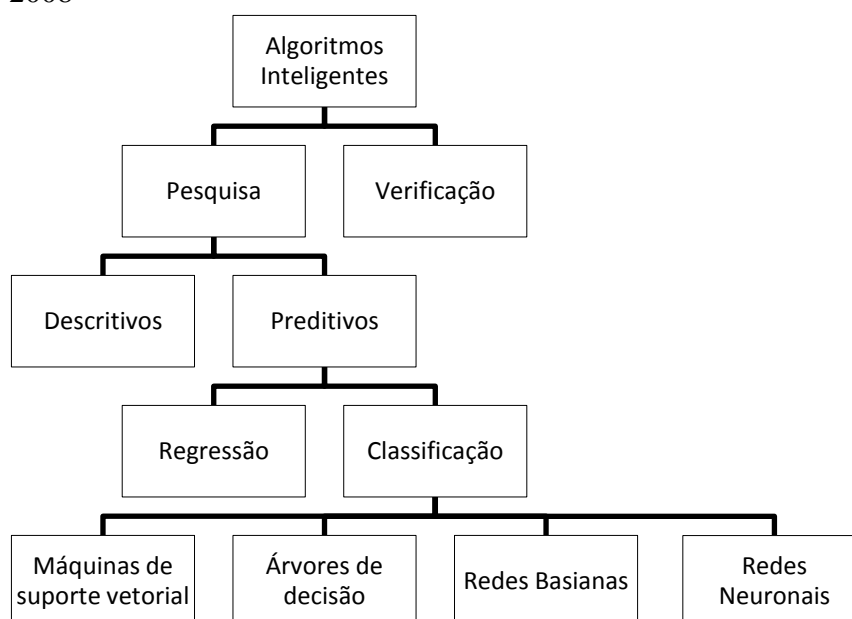
### 2.1. Algoritmos Inteligentes

Provenientes da designação *Data Mining* utilizada pela comunidade científica internacional, os algoritmos inteligentes correspondem à ciência e tecnologia de exploração de dados por forma a descobrir, previamente à sua utilização, padrões desconhecidos (Rockach e Maimon, 2008). Na generalidade, os algoritmos inteligentes são baseados numa aprendizagem indutiva, onde o modelo é construído implícita ou explicitamente pela generalização de um número aceitável de áreas de treino. O pressuposto na abordagem indutiva é que o modelo construído é aplicado no futuro a dados desconhecidos (Rockach e Maimon, 2008).

Os algoritmos inteligentes podem ser orientados com base em dois métodos distintos: orientados por verificação ou pesquisa (Figura 9). A orientação por verificação avalia a hipótese proposta pelo supervisor através dos métodos mais comuns de estatística, como por exemplo o t-teste das médias ou a variância. A orientação por pesquisa identifica automaticamente padrões nos dados em análise, e podem por sua vez envolver dois métodos:

- i. Descritivos: focam-se na perceção da forma como os dados de relacionam;
- ii. Preditivos: constroem um modelo do comportamento dos dados para preverem uma ou mais variáveis relacionadas com as áreas de treino.

Figura 9 Representação esquemática dos tipos de abordagens adaptada de Rockach e Maimon, 2008



Como apresentado no esquema da Figura 9 os métodos preditivos podem recorrer a diferentes modelos. Por modelo é entendida a relação entre as variáveis de entrada (independentes) e as variáveis de saída (dependentes). Os modelos de regressão, dadas as variáveis de entrada, preveem-nas num dado domínio de valores reais, enquanto que os modelos de classificação realizam a previsão em classes prédefinidas.

## **2.2. Árvores de Decisão em classificação de padrões**

As árvores de decisão são um modelo preditivo que podem ser utilizadas como modelo de regressão e de classificação. Nas situações em que a árvore de decisão utiliza o modelo de regressão designa-se por árvore de regressão, e por árvore de classificação aquando do uso do modelo de classificação.

A árvore de decisão apresenta uma estrutura hierarquia onde o primeiro nó ou nó da raiz é representado no topo da árvore, conectado por ligações ou ramos a outros nós. Existe uma sucessiva ligação de nós até que se atinja os nós terminais ou nós folhas, que não têm ligações de descendência.

Considerando o esquema apresentado na Figura 9 como uma esquematização de árvore, o nó “Algoritmos Inteligentes” corresponde ao nó raiz, ligado sucessivamente a outros nós. Os algoritmos “Máquinas de suporte vetorial”, “Árvores de Decisão”, “Redes Basianas”, e “Redes Neurais” correspondem aos nós terminais.

A classificação de um determinado padrão começa no nó raiz, onde se questiona o valor de um determinado atributo do padrão em estudo. As diferentes ligações descendentes a partir do nó raiz correspondem aos diferentes valores do atributo selecionado. Mediante a resposta do padrão ao valor do atributo, segue-se a ligação até ao nó descendente. O processo repete-se sucessivamente até alcançados os nós terminais.

Os atributos são armazenados numa D-tupla, com D igual ao número de atributos. No presente trabalho são utilizados quatro atributos, relativos aos valores digitais das quatro bandas utilizadas: 3 bandas da região do visível e a banda do infra-vermelho próximo (4-tupla).

### **2.2.1. Critérios de Divisão**

De acordo com o princípio fundamental da ciência, conhecido como *Occam's razor*, quando se procura uma explicação para qualquer fenómeno, devem ser realizadas o menor número possível de suposições e eliminar aquelas que não fazem diferença na previsão observada das hipóteses explicativas (Rockach e Maimon, 2008). Assim, são

preferíveis as decisões que levam a árvores simples e compactas com poucos nós (Duda *et al.*, 2001).

Para a criação da árvore ótima, no nó N procuramos o atributo que proporcione a obtenção de nós descendentes o mais “puros” possíveis. Na prática é mais conveniente definir impureza de um nó,  $i(N)$ , em detrimento de pureza.

O principal objetivo é alcançar o grau de impureza igual a zero, de modo a que os padrões que se enquadram no nó N pertençam todos à mesma classe (c). Existem várias formas de determinar a impureza num determinado nó, sendo as mais frequentemente utilizadas (Duda *et al.*, 2001):

1. Entropia:

$$i(N) = - \sum_j P(c_j) \log_2 P(c_j) \quad (1)$$

2. Índice de Gini:

$$i(N) = 1 - \sum_j P^2(c_j) \quad (2)$$

3. Erro de classificação:

$$i(N) = 1 - \max_j P(c_j) \quad (3)$$

onde  $P(c_j)$  é a fração das variáveis independentes no nó N que pertence à classe  $c_j$ .

Como referido anteriormente, se os padrões num determinado nó pertecerem todos à mesma classe então a impureza é zero, caso contrário é positiva com valor máximo quando classes estão igualmente representadas  $P(c_j) = P(c_i)$ .

É escolhido o valor  $s$  do atributo T que diminui o máximo possível a impureza. Por exemplo, na banda do infravermelho próximo (T) seleccionar os pixels com valor digital inferior a 10 ( $s$ ). A diminuição da impureza é designada por ganho de informação e é expressa por (Duda *et al.*, 2001):

$$\Delta i(N) = i(N) - P_L i(N_{Left}) - (1 - P_L) i(N_{Right}) \quad (4)$$

Sendo  $P_{Left}$  a fração de dados do nó N que se move para o nó descendente da esquerda.

Como os valores do padrão são números reais (valores digitais dos pixels) é importante utilizar algoritmos descendentes para determinarem o melhor plano de divisão. Nas situações em que dois valores  $s$  do atributo T obtiverem o mesmo ganho de informação, o valor utilizado como critério de divisão deve corresponder à média simples ou pesada.

Até ao momento foi assumido que a classificação se processa para duas classes. Contudo, em muitos cenários as variáveis independentes são agrupadas em mais do que duas classes. O critério *twoing* é útil para o desenvolvimento de árvores de decisão binárias mas com multiclases. As variáveis independentes são agrupadas em super classes C que contêm as variáveis representativas de várias classes,  $C = \{c_1, c_2, c_3, \dots\}$ ,



$c_j\}$ . O objetivo é maximizar o ganho da informação na divisão da superclasses  $C$  em  $C_1$  e  $C_2$ , e recorrendo à equação 4 o processo de seleção do valor  $s$  do atributo  $T$  passa a ser igual ao processo com duas classes.

O método de maximizar o ganho de informação com optimização das decisões localmente não garante que conduza a um resultado global óptimo, e que a árvore treinada corresponda à árvore mais compacta (Duda *et al.*, 2001).

A escolha da função de impureza raramente parece afetar o classificador final e a sua precisão. Na prática, o critério de paragem e o método de poda são mais importantes do que a função de impureza na determinação da precisão final do classificador (Duda *et al.*, 2001).

No presente o desenvolvimento da árvore de classificação é baseado no índice de Gini. Os atributos utilizados correspondem às bandas disponíveis das imagens, considerando a seguinte designação: V1 (banda do vermelho), V2 (banda do verde), V3 (banda do azul), e V4 (banda do infravermelho).

### 2.2.2. Critérios de Paragem

O critério de paragem estipula quando é que o indutor pára de dividir os nós, sendo o indutor o algoritmo de divisão de nós. Este critério pode ser baseado em vários parâmetros: após um número fixo de épocas, após um certo tempo de processamento, quando o erro mínimo da função é alcançado (minimização do erro da função selecionada) e à minimização do erro de generalização. O erro de generalização mede a capacidade do indutor generalizar as decisões para dados desconhecidos. Existem duas abordagens para determinar o erro de generalização: teórica e empírica.

Na abordagem teórica existe uma combinação entre o erro de treino do indutor e a capacidade em induzir diferentes classificadores para estimar o erro de generalização. Nesta combinação, um baixo erro de treino do indutor não garante um reduzido erro de generalização. A técnica Basiana, e parâmetros físicos estatísticos são dois exemplos (David Wolpert, 1995).

A abordagem empírica, em oposição à abordagem anterior não é baseada apenas em suposições, mas sim em observações. O método *holdout* consiste na divisão aleatória das áreas de treino em dois subconjuntos: aprendizagem e teste. O subconjunto de aprendizagem é utilizado pelo indutor para construir o classificador e é quantificada a taxa de classificações incorretas no subconjunto de teste. O erro no subconjunto de teste providencia uma melhor estimativa do erro de generalização, evitando o fenómeno de sobreajustamento (Rockach e Maimon, 2008).

### Validação cruzada $n$ -vezes

A validação cruzada é uma variante do método *holdout*. Em que as áreas de treino são aleatoriamente divididas em  $n$  subconjuntos mutualmente exclusivos de aproximadamente igual tamanho. O indutor é treinado e testado  $n$  vezes. Em cada iteração, é utilizado um diferente subconjunto para teste e  $n-1$  subconjuntos para treino. O subconjunto de teste é utilizado para estimar o erro de generalização, e como o objetivo final é alcançar o valor mínimo do erro de generalização, o classificador é treinado até atingir o erro mínimo de validação.

Seja  $err$  o erro de generalização desconhecido mas verdadeiro do classificador, e  $k$  o número de classificações incorretas no número total de áreas de treino, isto é, variáveis de independentes conhecidas ( $D$ ), o número de classificações incorretas apresenta uma distribuição binomial (Duda *et al.*, 2001):

$$P(i) = \binom{D}{i} err^i (1 - err)^{D-i} \quad (5)$$

Então, a fração de amostras de teste classificadas incorretamente é precisamente a máxima verosimilhança estimada para  $err$  em cada iteração:

$$err_n = \frac{k}{D} \quad (6)$$

Após a conclusão da validação cruzada  $n$ -vezes, o erro de generalização é expresso como média simples do erro de generalização em cada iteração ( $err_n$ ) (Rockach e Maimon, 2008; Hastie *et al.*, 2009) :

$$err = \frac{1}{n} \sum_{i=1}^n err_i \quad (7)$$

## **2.3. Poda da Árvore de Decisão**

Utilizando os critérios de paragem como meio para alcançar a árvore de classificação ótima, por vezes ocorre o efeito horizonte. Este efeito consiste numa paragem da divisão dos nós antes do momento ideal da paragem, conduzindo a uma árvore pequena (subajustamento das áreas de treino) ou numa paragem depois do momento ideal tendo como consequência um sobreajustamento das áreas de treino.

Na prática, no subajustamento os nós terminais são localizam-se junto ao nó raiz, beneficiando a exatidão global (aparente) do classificador mas bloqueando a divisão de benéfica de possíveis nós descendentes. No sobreajustamento, o classificador torna-se demasiado específico para classificar as áreas de treino e pouco eficiente a efetuar a generalização para dados desconhecidos.

Uma alternativa aos critérios de paragem para a obtenção da árvore de decisão ótima é a utilização do método de poda das árvores. Breiman *et al.* (1984) desenvolveram a

metodologia baseada em critérios de paragem alargados que permitem que a árvore se sobreajuste às áreas de treino. Em seguida, num processo reversivo, as sub-ligações da árvore sobreajustada que não contribuem para a precisão da generalização do classificador são removidas. Todos os pares de nós terminais vizinhos são considerados para eliminação. Qualquer par de nós terminais considerados que satisfaçam um pequeno aumento na impureza são eliminados, e o seu nó ancestral passa a ser classificado como nó terminal.

Uma vantagem em utilizar a poda das árvores de decisão é que evita o efeito horizonte, e como não existe a necessidade de guardar uma percentagem de áreas de treino para testar o classificador como na validação cruzada, a poda da árvore recorre a todas as áreas de treino. A poda também permite tornar a árvore mais eficiente ao compactar várias regras de decisão de vários nós numa regra mais simples a ser implementada num único nó. Contudo, a execução da poda exige um custo computacional enorme sendo inoperacional para quantidades de dados muito elevados.

Várias técnicas de poda foram desenvolvidas e um nó é podado se esta ação melhorar ou cumprir um certo critério. As técnicas *top-down* inciam a tarefa no nó raiz e seguem as ligações até aos nós terminais. As técnicas *bottom-up* iniciam a atividade nos nós folhas e reversivamente seguem as ligações até ao primeiro nó.

Para analisar a técnica de poda e a sua potencialidade no desenvolvimento de árvores de classificação com o fim de produzir um mapa temático onde são representadas as classes da superfície da terra: Mar, Sedimentos em Suspensão (SS), Zona de Rebentação (ZR), Areia Húmida (AH), Areia Seca (AS), foi definido o parâmetro de complexidade (cp). O cp é a diferença do erro do classificador antes e depois de executar a divisão de um determinado nó.

Foram definidos dois valores de cp: 0 e 0.01. Quando o indutor dividir um nó e o erro não diminuir, ou não diminuir mais do que 0.01 unidades, então o indutor pára de desenvolver a árvore de classificação. Em simultâneo foram utilizadas zero e 5 validações cruzadas para determinar o erro de generalização respetivamente para os valores de 0 e 0.01 do cp.

A análise dos resultados provenientes da variação dos valores de poda será efetuada nas duas aplicações do presente trabalho: a classificação de praias e a extração de formas/objetos costeiros.

## **2.4. Avaliação da Precisão da Classificação**

Para avaliar a precisão de um mapa temático de ocupação do solo são utilizados os métodos de análise estatística multivariada discreta. Estes métodos são utilizados para avaliar a concordância entre as distribuições estatísticas de variáveis que

constituem os atributos (resposta espectral) dos pixéis e que são alvo de uma classificação cruzada realizada sobre uma amostra desses pixéis (Fonseca e Fernandes, 2004). A classificação cruzada gera uma tabela de contingências (matriz de erro ou matriz de confusão) que apresenta as enumerações de todas as possíveis combinações de avaliações dos atributos.

A estatística mais simples que se pode determinar a partir da matriz de erro é a percentagem de elementos correctamente classificados (exatidão global ou precisão do conjunto) na imagem, em que é igual à razão entre o número de pixéis correctamente classificados e o número total de pixéis.

A exatidão global (EG) pode ser determinada para conjunto, designando-se por exatidão do produtor. Esta exatidão é definida pelo número de pixéis correctamente classificados na classe e o número total de pixéis classificados nessa classe. Esta medida indica se as áreas de treino da classe foram bem identificadas.

A exatidão do utilizador é calculada dividindo o número total de pixéis correctamente classificados numa classe pelo número de pixéis que foram atribuídos a essa classe. Esta medida indica qual a probabilidade de um pixel classificado numa dada classe representar, realmente, essa classe no solo (Lillesand e Kiefer, 2000).

Outro método para avaliar a precisão da classificação é o de avaliar a concordância entre as distribuições das variáveis aleatórias que representam as duas medições das respostas espectrais, ou seja, entre as distribuições multinomiais das respostas espectrais da amostra, quando considerado como referência e quando considerada como teste.

O resultado desta avaliação é expresso numa variável K (*Kappa*).

- $Kappa = 0$ , a concordância real é igual à concordância aleatória;
- $Kappa > 0$ , a concordância real é maior do que a concordância devido ao acaso;
- $Kappa = 1$ , a concordância é perfeita.

### 3. Resultados e Discussão

### 3.1. Classificação de Praias

A árvore de classificação desenvolvida pelo algoritmo CART (Therneau *et al.*, 2012), com base nas áreas de treino fornecidas é apresentada esquematicamente na Figura 10. A respetiva rotina desenvolvida é apresentada no Anexo I, secção B. A Figura 11 apresenta a árvore de classificação de praias, mas com requisito de (cp) 0.01 para melhoria da impureza para a divisão de um determinado nó.

Como descrito no segundo capítulo, no pacote *Rpart* do R a função *rpart* apresenta a capacidade de elaboração dos critérios de decisão assim como a capacidade de terminar o processo de fragmentação dos conjuntos de dados de treino.

Figura 10 Árvore de classificação de praias

### Árvore - classificação praias

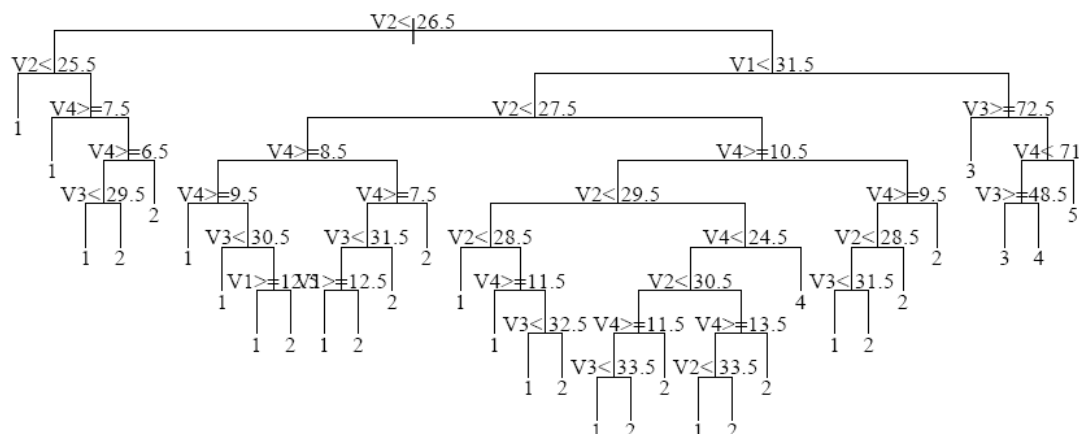
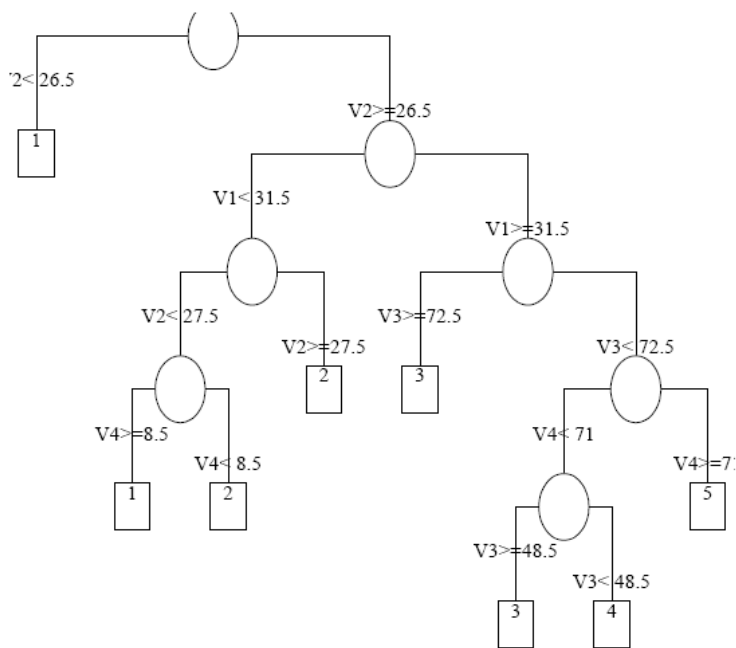


Figura 11 Árvore de classificação podada para classificação de praias



A árvore de classificação ilustrada na Figura 10 apresenta um total de 30 descentes, para classificar as imagens em 5 classes pré-definidas. Na Figura 11 a árvore de classificação podada apresenta apenas 7 nós descentes.

Para analisar o desenvolvimento das árvores, apresentam-se as precisões para ambas as árvores nas Tabelas 3 e 4.

Tabela 3 Precisões para árvore de classificação de praias

Classe	EG	Kappa	Produtor	Utilizador
Mar	98.18534	0.970924	97.76161	98.64483
SS			98.50575	97.71950
ZR			95.20548	98.58156
AH			97.14286	89.47368
AS			99.76471	98.83450

Tabela 4 Precisões da árvore podada de classificação de praias

Classe	EG	Kappa	Produtor	Utilizador
Mar	96.87878	0.949966	96.83258	96.66855
SS			96.36776	96.80730
ZR			95.20548	98.58156
AH			96.87500	81.57895
AS			99.76471	98.83450

## Resultados e Discussão

O classificador (Tabela 3) apresenta uma precisão global muito boa (98.2%), demonstrada pela exatidão da classificação individual de cada classe. A classe com menor exatidão foi a zona de rebentação. É espectável que esta classe apresentasse o valor mais baixo da precisão individual das classes devido à sua elevada variabilidade espectral.

Na zona de rebentação, a água pode apresentar outros elementos na sua composição como bolhas, espuma e sedimentos em maior concentração do que nas restantes zonas da imagem. Assim, a resposta espectral é mais ampla, sendo mais complexo a delimitação da fronteira para esta classe.

É ainda importante salientar que a classe Areia Húmida apresenta uma exatidão de utilizador de 89.5%. Em Teodoro *et al.*(2011) podemos verificar através da frequência dos valores digitais para cada classe, que a classe Areia Húmida apresenta valores digitais comuns às classes Areia Seca e Zona de Rebentação. Esta partilha de alguns intervalos da escala dos valores digitais [0,255] origina uma baixa precisão do utilizador para a classe Areia Húmida.

Comparando as exatidões do classificador com e sem a poda, verifica-se que ao podar a árvore de classificação o classificador perde precisão e sensibilidade.

Praticamente todas os parâmetros estatísticos avaliados na Tabela 3 e 4 baixam quando se procede à poda da árvore de classificação. Durante a poda da árvore, os nós que não reduzam a impureza em pelo menos 0.01 unidades são eliminados, e o seu ancestral passa a ser o nó terminal. Contudo, verifica-se que existem nós que reduzem o erro em 0.003007675 (Anexo I, secção C).

A partir de um determinado nível da árvore, a redução da impureza torna-se menos expressiva, admitindo valores muito reduzidos nos nós intermédios próximos dos nós terminais. Analisando as matrizes de confusão das duas árvores (Tabela 6 e 7), concluímos que a perda de precisão reflete-se principalmente na perda de sensibilidade em classificar a classe Mar e Sedimentos em Suspensão. De acordo com o apresentado no capítulo I, os corpos de água apresentam uma variação da resposta espectral muito reduzida, pelo que proceder à classificação de corpos de água é uma tarefa bastante complexa.

Se durante a poda são eliminados os critérios que possibilitam a distinção eficaz entre a classe Mar e Sedimentos em Suspensão (Tabela 5 e 6), o classificador resultante desta função terá menor sensibilidade para classificar corretamente os pixels como Mar e Sedimentos em Suspensão. Embora para o limite, com  $cp=0.01$ , o respetivo erro relativo esteja próximo de zero (Figura 12), para  $cp=0.001$  o erro relativo é menor do que para  $cp=0.01$ .

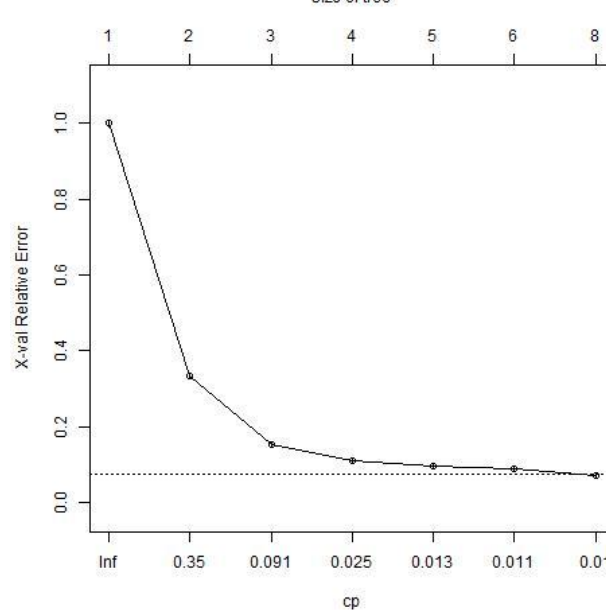
Tabela 5 Matriz de confusão para árvore podada de classificação de praias

Com poda					
	Mar	SS	ZR	AH	AC
MAR	1712	59	0	0	0
SS	56	1698	0	0	0
ZR	0	1	139	1	0
AH	0	4	2	31	1
AS	0	0	5	0	424

Tabela 6 Matriz de confusão para árvore de classificação de praias

Sem poda					
	Mar	SS	ZR	AH	AC
MAR	1747	24	0	0	0
SS	40	1714	0	0	0
ZR	0	1	139	1	0
AH	0	1	2	31	1
AS	0	0	5	0	424

Figura 12 Critério de paragem vs erro relativo



As Figuras 13 e 14 ilustram as praias classificadas por classificador não podado e podado respetivamente. As Tabelas 8 e 9, apresentam a percentagem das classes presentes em cada uma das imagens classificadas. A rotina desenvolvida para a composição das imagens encontra-se no Anexo I, secção D.

As diferenças centram-se na quantidade de pixels classificados como Mar, Sedimentos Suspensos e Areia Húmida. A classe Areia Húmida, como analisado anteriormente, apresenta uma precisão de utilizador razoável, pelo que uma variação na precisão do classificador origina uma classificação diferente de alguns pixels previamente classificados como Areia Húmida.

A principal alteração é entre as classes Mar e Sedimentos Suspensos. Na imagem criada pelo classificador podado o número de pixels classificados como Mar diminuem e



## Resultados e Discussão

aumenta o número de pixels da classe Sedimentos em Suspensão. Numa análise visual das imagens classificadas verificamos que em mar aberto existem mais pixels dispersos classificados como sedimentos suspensos na imagem proveniente do classificador podado (Figura 11) do que na imagem proveniente do classificador sem poda (Figura 10). Este aumento de pixels é inconsistente como o verificado na realidade.

Tabela 5 Percentagem de cada classe na imagem

Sem poda		
Classe	Pixeis	Percentagem
Terra	309499	20.05
Mar	849671	55.03
SS	321006	20.79
ZR	16294	1.06
AH	9902	0.64
AS	37603	2.44

Tabela 6 Percentagem de cada classe na imagem

Com poda		
Classe	Pixeis	Percentagem
Terra	309499	20.05
Mar	845117	54.74
SS	327006	21.18
ZR	16294	1.06
AH	8456	0.55
AS	37603	2.44

Em Teodoro *et al.* (2010), a classificação de praias através de redes neuronais embora apresente precisões idênticas às árvores de decisão (sem poda da árvore), com uma exatidão global de 98.6%, apresenta uma classificação mais sensível às correntes de retorno, ou seja, não existe a classificação de pixels como sedimentos em suspensão em mar aberto.

Os métodos de classificação clássicos aplicados à mesma área de estudo apresentam precisões inferiores às árvores de decisão, com o classificador máxima verosimilhança e paralelepípedo a obterem as precisões mais próximas das árvores de decisão com 95% exatidão global (Teodoro *et al.*, 2010). Contudo, as árvores de decisão apresentam uma sensibilidade semelhante a estes classificadores para classificar os sedimentos em suspensão. Em ambos os métodos, as correntes de retorno não são definidas com uma precisão tão elevada como nas redes neuronais.

Figura 13 Imagem classificada

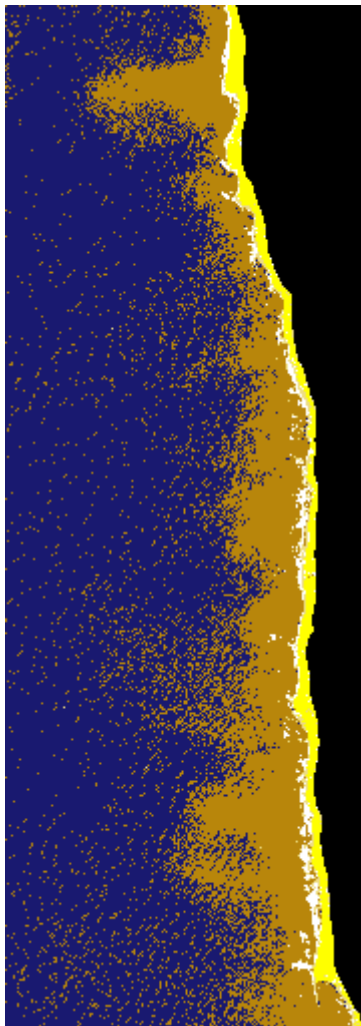
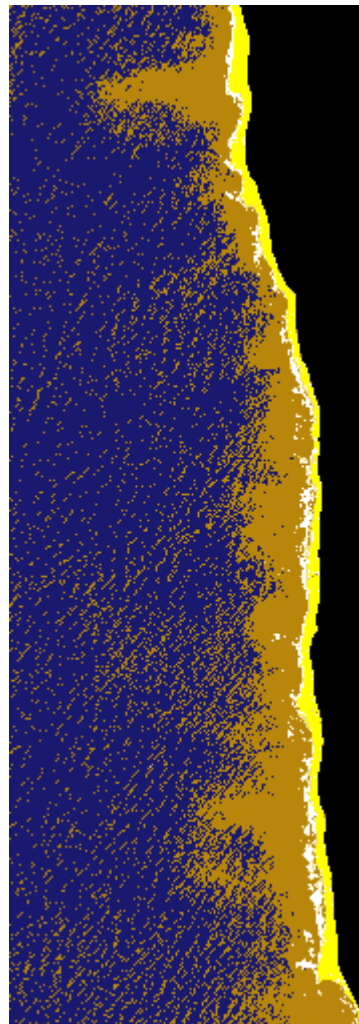


Figura 14 Imagem classificada com poda (cp=0.01 e xval=5)

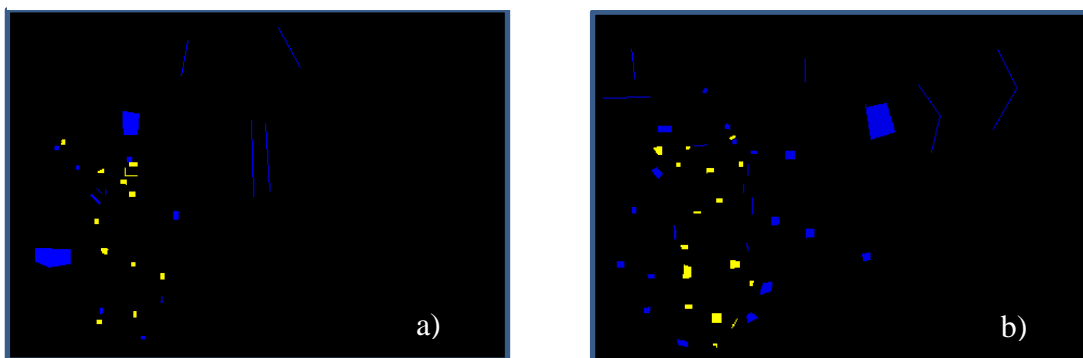


Na aplicação das árvores de decisão a grandes volumes de dados para processamento, a poda é uma abordagem eficaz na otimização de tempo de processamento, contudo existe uma perda de precisão do algoritmo. Dependendo das áreas de estudo esta perda de precisão pode ou não ser relevante. No presente trabalho, dada a reduzida variação da resposta espectral é importante manter o máximo grau de precisão possível do classificador.

### 3.2. Extração de formas costeiras

A segunda aplicação das árvores de decisão sob estudo neste trabalho é a extração de formas costeiras, mais concretamente a restinga do rio Douro. Foram utilizadas 6 imagens IKONOS-2 entre o ano de 2001 e 2007. Devido à constante alteração da forma da restinga, para cada ano foram definidas áreas de treino no *software* PCI Geomatica V8.2 (Figura 15). As áreas de treino relativas aos restantes anos são apresentadas no Anexo II, secção A.

Figura 15 Áreas de treino da restinga. a) dezembro de 2001; b) junho de 2005



A separabilidade das classes para cada um dos anos de estudo, medidas através da distância Bhattacharyya (Duda *et al.*, 2001), encontram-se apresentadas na Tabela 9.

Tabela 7 Separabilidades das áreas de treino

Ano	Mês	Separabilidade	
		Restinga	Não-restinga
2001	dezembro	1.946270	
2004	junho	1.927209	
2004	julho	1.923372	
2005	junho	1.997407	
2005	setembro	1.999138	
2007	junho	2.000000	

As duas imagens de 2005 e 2007 apresentam uma separabilidade muito boa, enquanto nas imagens de 2001 e 2004 a separabilidade é um pouco mais baixa. Embora a separabilidade destas últimas 3 imagens seja aceitável é um indício de que a definição da fronteira de cada classe não será exclusiva. A extração da forma deste tipo de corpos

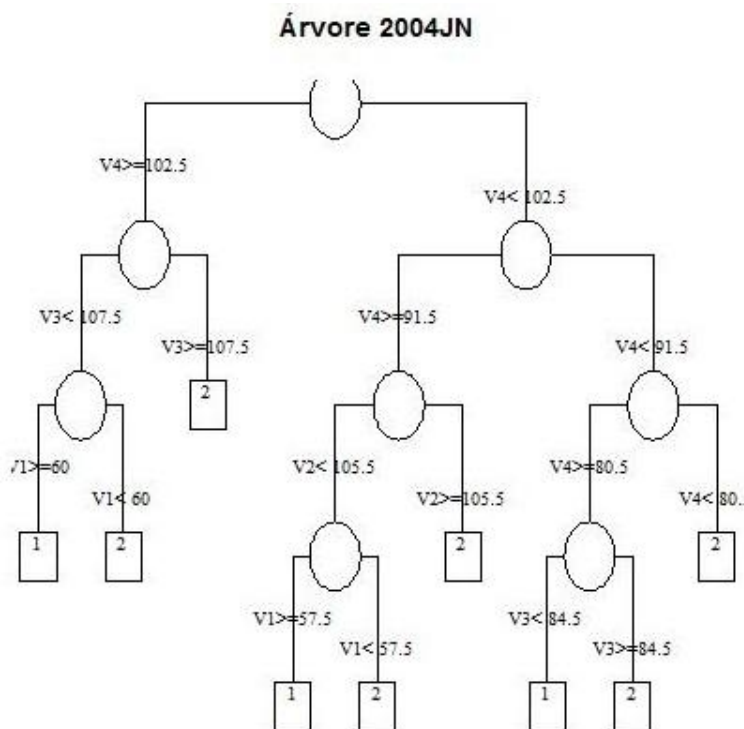
## Resultados e Discussão

arenosos é complexa devido à não exclusividade de áreas de cada classe. Esta não exclusividade deve-se à existência de baixios no limite do corpo arenoso, que embora pertençam ao corpo de água, a sua profundidade é mínima e permitem a reflexão da radiação pelos sedimentos do fundo. Assim, a fronteira entre a restinga e o corpo de água é muito tenue.

A rotina que extrai os vetores das imagens das áreas de treino faz parte integrante do Anexo II, secção B. A rotina para a leitura dos vetores e desenvolvimento das árvores de classificação é apresentada no Anexo II, secção C.

Para cada um dos anos, e respetivos meses, foram desenvolvidas as árvores de decisão com base nas respetivas áreas de treino. São apresentadas nas Figuras 16, 17 e 18 as árvores relativas aos meses de junho do ano de 2004, 2005 e 2007. As restantes árvores serão apresentadas no Anexo II, secção D.

Figura 16 Árvore de classificação para as imagens de junho de 2004



Como referido anteriormente, a zona do estuário é uma área particular na interação da radiação entre a água e os sedimentos nela contidos. A definição dos critérios de divisão são complexos quando se trata de corpos marinhos, especialmente estando enquadrados numa região de depósito final de sedimentos como é o caso do estuário em estudo.

Posteriormente as árvores desenvolvidas foram sujeitas a um critério de paragem, para verificar se o processo de classificação se tornava mais eficiente. Para tal, o desenvolvimento dos critérios de divisão ficaram sujeitos ao critério de  $cp=0.01$ . Ou

seja, para proceder à divisão de um determinado nó, este tem de reduzir a impureza em pelo menos 0.01 unidades.

Figura 17 Árvore de classificação para as imagens de junho 2005.

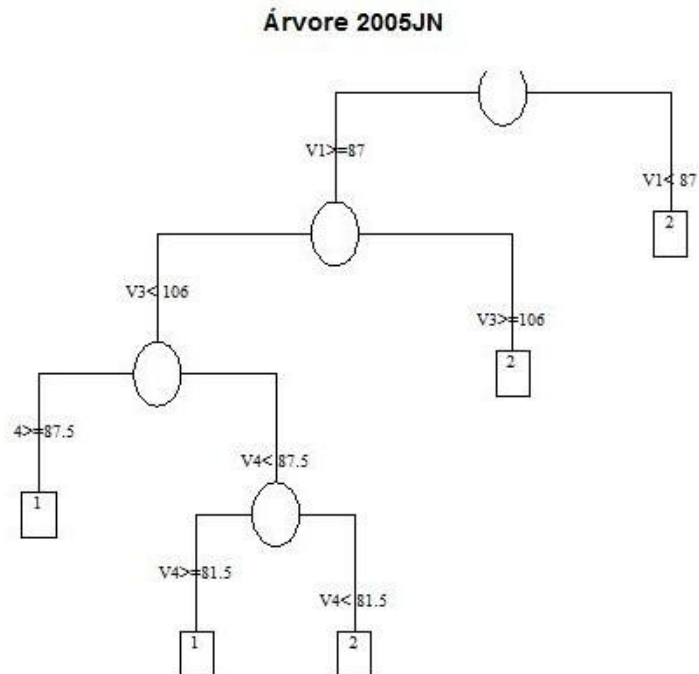
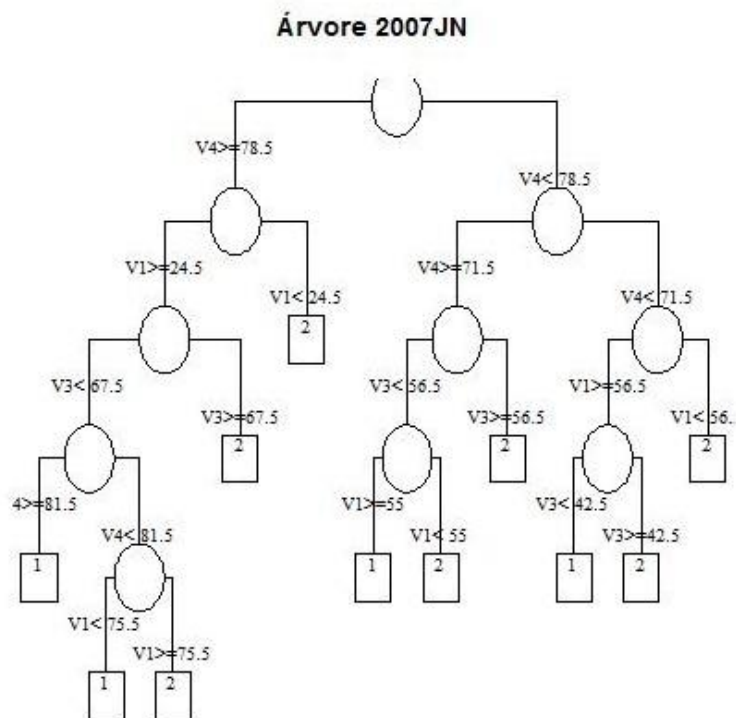


Figura 18 Árvore de classificação para as imagens de junho de 2007



Os resultados do critério de paragem para  $cp=0.01$  são apresentados nas Figuras 18, 19 e 20. Apenas as árvores de junho de 2004, 2005 e 2007 foram simplificadas.

Figura 19 Árvore de decisão podada para as imagens junho de 2004

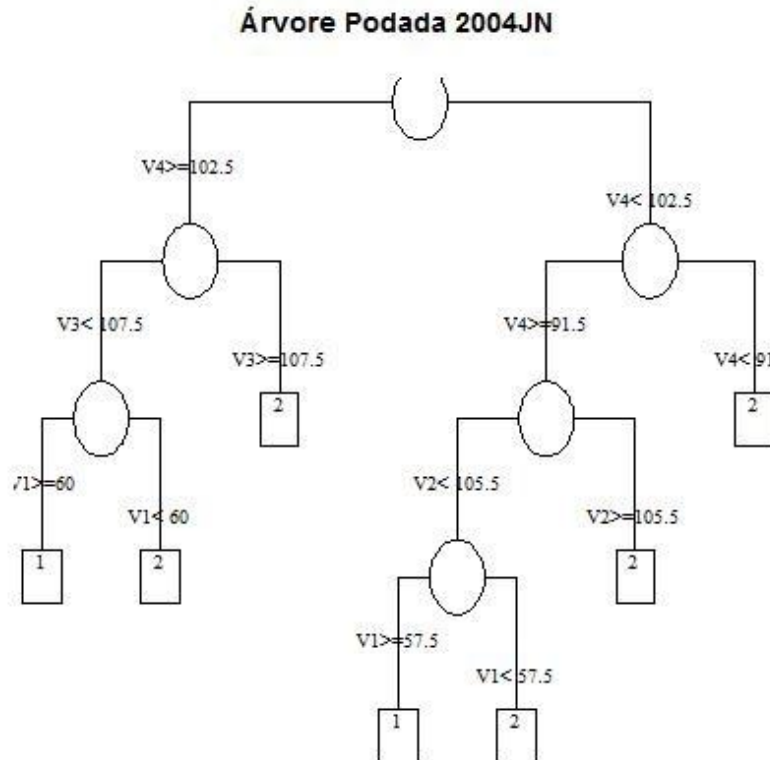


Figura 20 Árvore de decisão podada para as imagens de junho de 2005

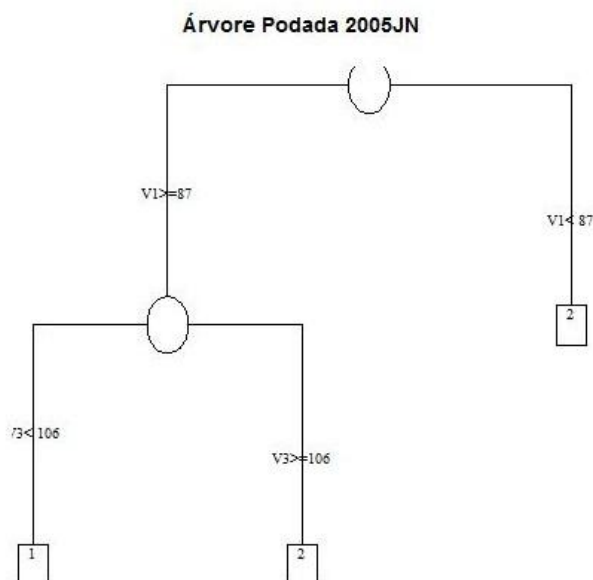
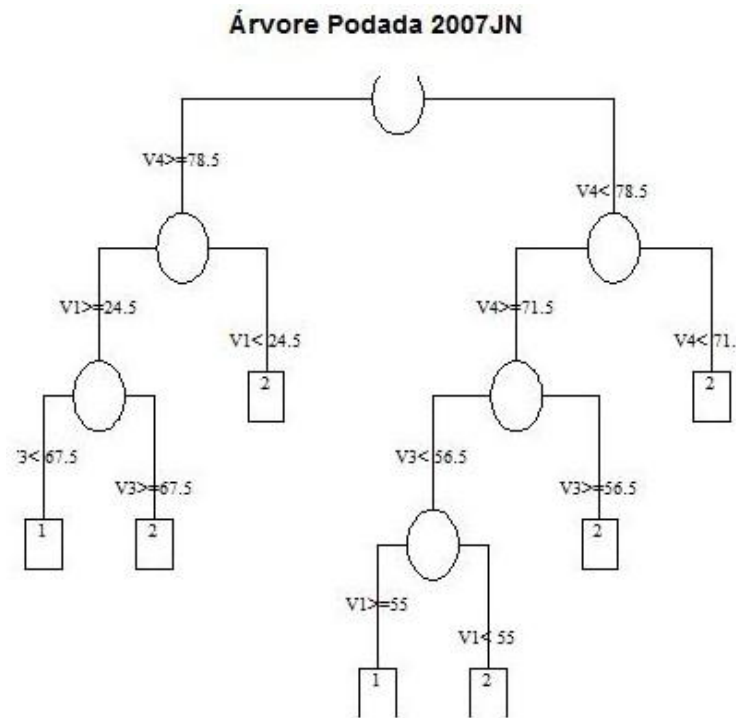
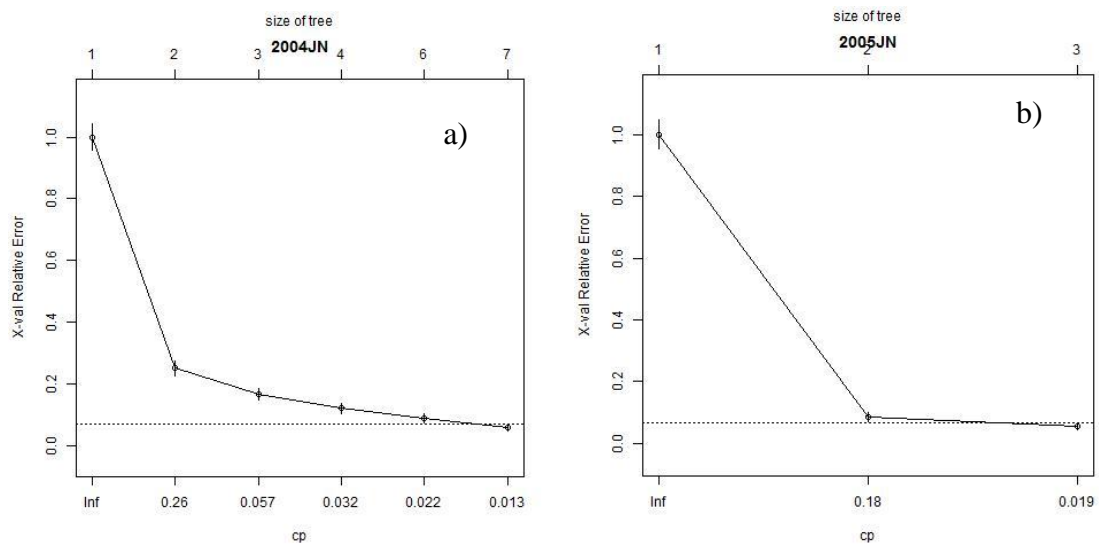


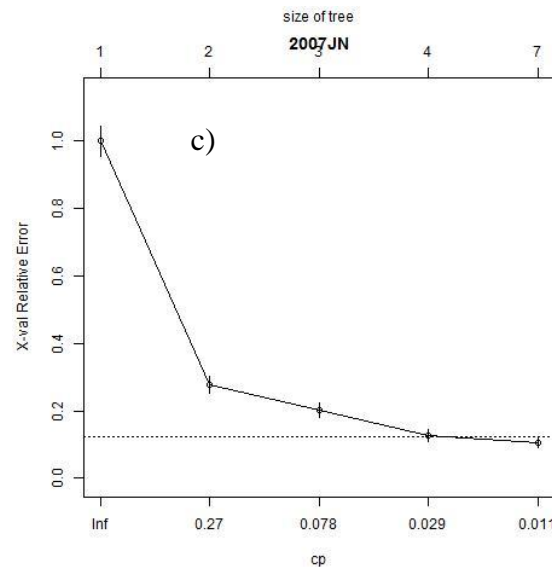
Figura 21 Árvore de decisão podada para as imagens de junho de 2007



Os erros relativos em função do critério de redução da impureza são apresentados nos gráficos das Figuras 19, 20, e 21 para os anos de 2004 (junho), 2005 (junho) e 2007 (junho), respectivamente.

Figura 22 Erro relativo em função do critério de paragem. a) junho de 2004; b) junho de 2005; c) junho de 2007





A utilização do critério de paragem tornam a árvore mais eficiente, reduzindo o número de nós a que os dados desconhecidos terão de se submeter para serem classificados. Contudo a aplicabilidade da eficiência atingida pela função compensa o tempo de processamento e complexibilidade computacional exigida pela função. No presente trabalho as árvores de decisão desenvolvidas sem a aplicação do critério de paragem apresentam poucos nós (por exemplo dois, 4, ou 5). Os dados desconhecidos a classificar não apresentam tamanhos muito grandes, pelo que o custo benefício da utilização da função deve ser sempre equacionado. De notar que os investigadores em árvores de decisão utilizam o termo de grandes quantidade de informação como terabyts de dados.

Na secção 3.1. verificamos que a poda ou limitação do desenvolvimento da árvore causa uma perda de precisão do classificador, devido ao número de nós que a árvore apresenta. Contudo, nesta secção apenas são consideradas duas classes pelo que as árvores apresentam poucos nós e a redução da impureza não alcança valores muito superiores à ordem de grandeza de 0.01. Face ao esforço computacional para a imposição do critério, neste caso em concreto a utilização do critério é desencorajada.

Tabela 8 Precisões do classificador para as seis imagens

Ano	Mês	EG	kappa
2001	dezembro	96.79634	0.89392
2004	junho	98.33055	0.95936
2004	julho	99.66330	0.99274
2005	junho	98.95833	0.96930
2005	setembro	100.00000	1.00000
2007	junho	98.94180	0.96965



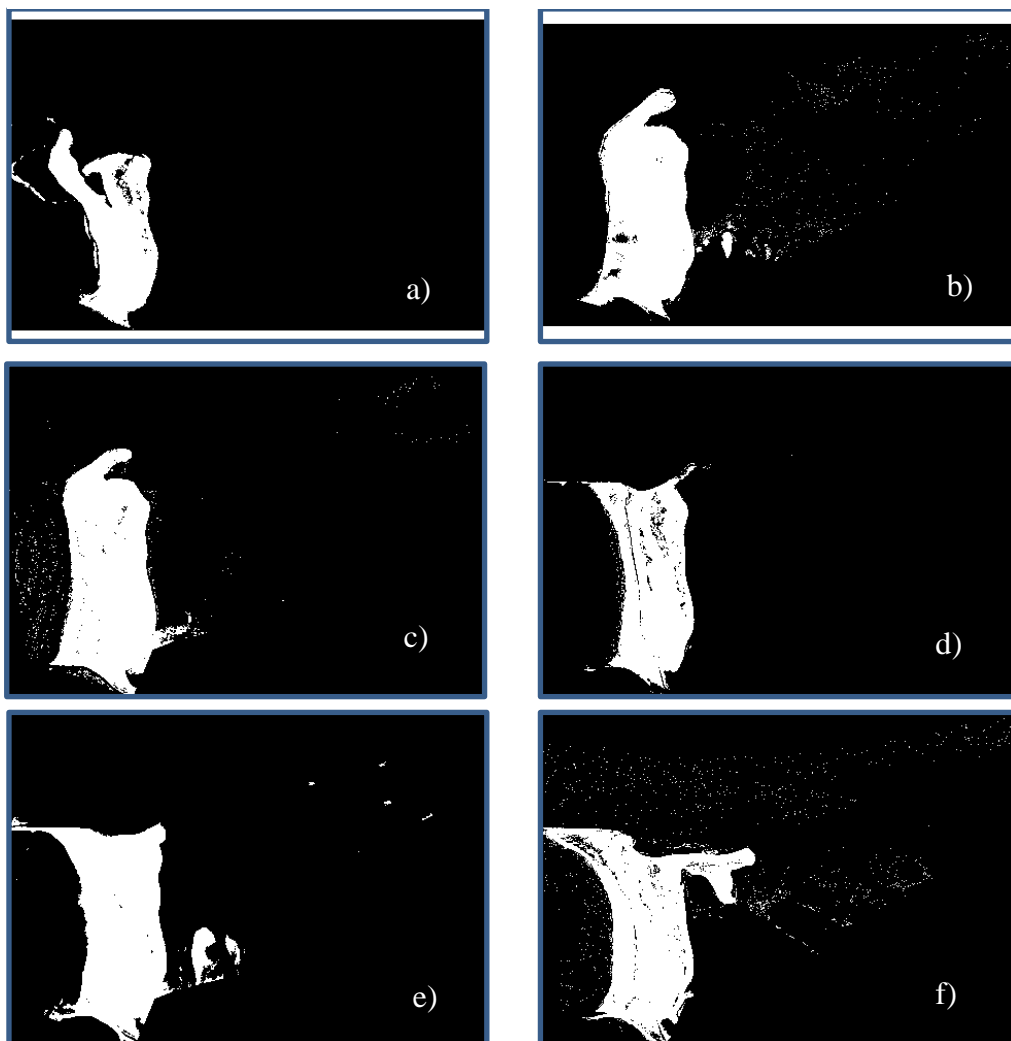
## Resultados e Discussão

O ano de 2001 apresenta os valores mais baixos da exatidão global e do parâmetro estatístico kappa. Estes resultados devem-se principalmente à presença de vegetação na zona norte da restinga. Deste modo a resposta espectral do local, é uma mistura entre a classe Cabedelo e “não” Cabedelo .

A representação das imagens classificadas processou-se no *software* Matlab® através da rotina que consta no Anexo II, secção E.

As imagens classificadas são apresentadas na Figura 20.

Figura 23 20 Imagens classificadas pelas árvores de classificação. a) dezembro de 2001; b) junho de 2004; c) julho de 2004; d) junho de 2005; e) setembro de 2005; f) junho de 2007.



É visível que o classificador nas imagens de 2004 e setembro de 2005, apresentaram dificuldades na separação entre a zona de alta concentração de sedimentos a Este da restinga e o próprio corpo arenoso. Esta confusão na classificação de pixels de sedimentos em suspensão com o próprio corpo arenoso (cabedelo) é devido à forte acumulação e concentração de sedimentos nas águas nesta região. Com o aumento da área do cabedelo localizado a Este do Cabedelo torna-se propício a acumulação de

sedimentos devido à fraca agitação marítima, e à redução da velocidade do caudal do rio e ainda devido à baixa profundidade.

Para a obtenção das imagens classificadas livres de ruído ou pixels mal classificados, foi realizada uma etapa de pós-processamento que consistiu em duas ações. A primeira a etapa baseou-se na eliminação dos pixels que não estão conectados ao corpo principal do Cabedelo. A segunda etapa consistiu no preenchimento do interior do cabedelo, onde o classificador tinha classificado pixels como “não” Cabedelo. As imagens classificadas com pós-processamento são apresentadas nas Figuras 24 à 29.

Figura 24 Forma do cabedelo em dezembro de 2001



Figura 25 Forma do cabedelo em junho de 2004



Figura 26 Forma do cabedelo em julho de 2004



Figura 27 Forma do cabedelo em junho de 2005



Figura 28 Forma do cabedelo em setembro de 2005



Figura 29 Forma do cabedelo em junho de 2007



A metodologia para o pós processamento permite compor a forma interior do Cabedelo, através da classificação de pixels no interior do polígono (Cabedelo). Contudo os limites do cabedelo são estabelecidos durante a classificação das imagens.

A fronteira entre o corpo arenoso do Cabedelo e da água é impreciso, dependendo inclusive da maré, caudal do rio, entre outros. Na região a Oeste do Cabedelo, região exposta à agitação marítima, a extração do limite é de extrema complexidade devido à rebentação das ondas. A precisão da extração da fronteira é o principal argumento na determinação da dimensão do Cabedelo.

Entre 2001 e 2007 o corpo do Cabedelo mudou significativamente. Em 2001, o corpo apresentava uma forma com pouca coesão, encontrando os seus limites, principalmente na extremidade livre, pouco definidos e com muitas saliências. Em 2004, a restiga que se encontrava muito saliente em 2001 migrou para o interior do estuário e formou um corpo mais denso (com menos saliências). Em setembro de 2005, o Cabedelo encontrava-se com maiores dimensões, devido à acumulação de sedimentos na região Este (interior do estuário). Em 2007, com a construção do molhe, o corpo arenoso ganhou maior dimensão na região norte (zona de ligação do Cabedelo ao molhe) e continuou a aumentar a sua dimensão na região Este (situação verificada em 2005).

Embora a forma do Cabedelo esteja em constantes alterações, a construção do molhe em 2007 permitirá estabilizar o corpo aneroso na sua globalidade.

Figura 30 Ampliação da zona oeste do cabedelo, em junho de 2007



A área do Cabedelo para as seis imagens são apresentadas na Tabela 11.

Tabela 9 Área do Cabedelo (m<sup>2</sup>) (adaptado de Teodoro e Gonçalves, 2012)

Data (dd-mm-aaaa)	Área			
	Árvores de decisão	GThE	Referência Manual	Referência dGPS
24-12-2001	173040	188123	191056	n.a.
03-06-2004	261376	260448	275761	265200
31-07-2004	261136	258973	260095	259864
03-06-2005	202048	207214	212819	222636
18-09-2005	248160	229186	228092	228688
06-06-2007	265744	256747	267208	225237

O método GThE é uma metodologia proposta em Teodoro e Gonçalves (2012), e é a metodologia que apresentou a melhor precisão para extração da forma do Cabedelo entre as restantes metodologias propostas (classificação baseado no pixel e orientada a objetos). Assim, neste trabalho é realizada uma comparação com dados de referência mas também ao método GThE.

As árvores de decisão apresentam uma boa aproximação no cálculo da área do Cabedelo, contudo verifica-se alguma discrepância com os dados de referência. Contudo esta mesma discrepância verifica-se entre os dados de referência, devido a fatores já inumerados como momento temporal de obtenção dos dados de referência, maré, agitação marítima, e vento. Os valores das áreas obtidos através das árvores de decisão estão dentro da ordem de grandeza da variação das áreas entre os dados de referência, pelo que podem ser consideradas como uma técnica válida para a extração e quantificação de formas costeiras.

No entanto, o método GThE apresenta uma melhor aproximação na quantificação da área do cabedelo. Este resultado é explicado pela melhor capacidade desta metodologia em diferenciar/extrair as fronteiras entre o Cabedelo e a água.

Contudo mais parâmetros podem ser incluídos no desenvolvimento das árvores de decisão. No futuro, devem ser analisados os resultados de árvores de decisão com mais variáveis de entrada, como a utilização de pesos para as classes, a utilização de diferentes combinações de bandas e a utilização de diferentes métodos de aprendizagem como validação cruzada  $n$  vezes.

Devido ao tempo inerente a um trabalho deste tipo não foi possível desenvolver em tempo útil de ser apresentado neste trabalho esses desenvolvimentos.

## 4. Conclusão

Neste trabalho foi abordada uma nova técnica para a análise do ambiente costeiro, como a classificação de praias e extração de formas de formas/corpos costeiros. Os estudos realizados nestas áreas apresentam um grande custo devido à quantidade de dados necessários para efetuar a análise da região costeira. Em alternativa, imagens de satélite e métodos de processamento de imagem assume um papel preponderante e de baixo custo.

As árvores de decisão aplicadas a imagens de alta resolução espacial obtiveram bons resultados na classificação de praias, e permitem determinar com precisão as correntes de retorno, importantes na quantificação da dinâmica sedimentar costeira. Esta técnica apresenta uma melhor capacidade de classificar estes ambientes do que as técnicas clássicas da detecção remota, como os algoritmos de máxima verosimilhança e paralelepípedo. Embora não produzam um resultado tão preciso como as redes neurais, as árvores de decisão são mais fáceis de implementar por utilizadores não peritos e mais fáceis de interpretar em relação a outras técnicas de *data mining*.

A extração de formas em ambiente costeiro apresenta complexidade devida à inexistência de uma fronteira bem definida entre os corpos/formas. As árvores de decisão conseguem obter uma extração do objeto com bastante precisão. A precisão obtida é da mesma ordem de grandeza que a precisão dos dados de referência. Dada a sua fácil interpretabilidade, as árvores de decisão assumem-se como uma metodologia alternativa à dos levantamentos de campo e à digitalização manual do objeto em estudo-técnica muito subjectiva e morosa.

Na extração de formas, dada a precisão do algoritmo GThE na determinação de formas e as áreas subjacentes, em trabalhos futuros deve ser considerada a utilização de uma banda de segmentação como atributo do algoritmo.

Em trabalhos futuros esta técnica pode ainda ser analisada com mais parâmetros, como pesos nas classes de objetos, diferentes abordagens à aprendizagem do classificador, e com diferentes atributos para análise, para aumentar a precisão do classificador e dos resultados obtidos.

Deste modo, este trabalho pretendeu ser uma primeira abordagem à aplicabilidade das árvores de decisão para estudos costeiros usando imagens de alta resolução espacial.

## 5. Bibliografia

- Bally, P., Béquignon, J., Arino, O., e Briggs, S. (2005). Remote sensing and humanitarian aid. *ESA Bulletin*, 132.
- Baptista, P., Bastos, L., Bernardes, C., Cunha, T., and Dias, J., (2008). “Monitoring sandy shores morphologies by DGPS - A practical tool to generate digital elevation models,” *J. Coast. Res.*, vol. 24, no. 6, pp. 1516–1528, 2008.
- Bird, E. (2008): “Coastal Geomorphology: An Introduction, second ed.” John Wiley & Sons.
- Breiman L., Friedman J., Olshen R., and Stone C., (1984). *Classification and Regression Trees*. Wadsworth Int. Group.
- Duda, R., Hart, P., Stork., D., (2001). *Pattern Classification*, second edition. John Wiley & Sons. Inc.
- Fernandes, J., e Fonseca, A. (2004). *Detecção Remota*. Lidel, Lisboa.
- Hastie, T., Tibshirani, R., Friedman, J., (2009). *The Elements of Statistical Learning – Data mining, Inference, and Prediction*, Second Edition. Springer series in statistics.
- Kiefer, R., e Lillesand, T. (2000) *Remote Sensing and Image Interpretation*. John Wiley & Sons, New York, 4, 21 pp.
- INAG (2000): “Plano Nacional da Água.” Instituto Nacional da Água
- NASA (2013). Remote Sensing: introduction and history.  
<<http://earthobservatory.nasa.gov/Features/RemoteSensing>>
- Pais-Barbosa, J., Veloso-Gomes, F., Taveira-Pinto, F., (2007). Coastal features in the energetic and mesotidal west coast of Portugal. *Journal of Coastal Research*, 50, pp. 459–463.
- Quinlan, J. Ross (1987) Simplifying decision trees, *International Journal of Man-Machine Studies*, 27, 221-234.
- Rockach, L., and Maimon, O., (2008). *Data mining with decision trees – theory and applications*. World Scientific Publishing Co. Pte. Ltd.
- Sunamura, T., (1988). Beach morphologies and their change. In: *Horikawa, K. (Ed.), Nearshore Dynamics and Coastal Processes*, University of Tokyo Press, pp. 136-166.
- Teodoro, A.C., Veloso-Gomes, F., Gonçalves, H. (2007). Retrieving TSM Concentration From Multispectral Satellite Data by Multiple Regression and

Bibliografia

- Artificial Neural Network. *IEEE Transactions on Geoscience and Remote Sensing*, vol. 45, no.5.
- Teodoro, A.C., Pais-Barbosa, J., Veloso-Gomes, F., Taveira-Pinto, F. (2009). Evaluation of Beach Hydromorphological Behaviour and Classification Using Image Classification Techniques. *Journal of Coastal Research*, vol 56.
- Teodoro, A.C., Pais-Barbosa, J., Veloso-Gomes, F., Taveira-Pinto, F., (2010). Identification of beach features/patterns through artificial neural networks techniques using IKONOS data. *ISPRS*.
- Teodoro, A.C., Pais-Barbosa, J., Veloso-Gomes, F., Taveira-Pinto, F., (2011). Identification of beach hydromorphological patterns/forms through image classification techniques applied to remotely sensed data. *International Journal of Remote Sensing*. Pp. 0143-1161.
- Teodoro, A.C., Gonçalves, H., (2012). A semi-automatic approach for the extraction of sandy bodies (sand spits) from IKONOS-2 data. *IEEE Transactions on Geoscience and Remote Sensing*, vol. 5, no.2.
- Therneau, T., Atkinson, E., (2012). An introduction to recursive partitioning using the RPART routines. *Mayo Foundation*.
- Veloso-Gomes, F., Taveira-Pinto, F., das Neves, L., Pais-Barbosa, J., (2006). EUrosion - A European Initiative for Sustainable Coastal Erosion. Pilot Site of River Douro – Cape Mondego and Case Studies of Estela, Aveiro, Caparica, Vale do Lobo and Azores, Porto, Portugal, 317 p.
- Wolpert, David H., (1995). *The Mathematics of Generalization: The Proceedings of the SFI/CNLS Workshop on Formal Approaches to Supervised Learning*. Perseus Publishing.
- Wright, L.D. and Short, A.D., (1984). Morphodynamic variability of surf zones and beaches: a synthesis. *Marine Geology*, 56, pp. 93-118.



## Anexos

### Anexo I

#### Secção A

```
% exportar vetor de pixeis para o R

%% Leitura das imagens
nome_pasta = 'D:\tese_mestrado\1_classificacao\imagens_editadas_8bt';

nome_fx_B1 = [nome_pasta, '\banda_B_ROI.tif'];
nome_fx_B2 = [nome_pasta, '\banda_G_ROI.tif'];
nome_fx_B3 = [nome_pasta, '\banda_R_ROI.tif'];
nome_fx_B4 = [nome_pasta, '\banda_NIR_ROI.tif'];
B1=imread(nome_fx_B1);
B2=imread(nome_fx_B2);
B3=imread(nome_fx_B3);
B4=imread(nome_fx_B4);

%% Exportar pixeis
% ler as bandas em colunas
B_all_coluna = [B3(:) B2(:) B1(:) B4(:)];
[nrl,nrc]=size(B_all_coluna);
nr_linhas = round(nrl/10);

%exportar as 4 bandas em 10 ficheiros separados
for k = 1:10
    if k < 10
        B_subset = B_all_coluna((1+(k-1)*nr_linhas):nr_linhas*k,:);
    else
        B_subset = B_all_coluna((1+(k-1)*nr_linhas):end,:);
    end

    nome_out = ['C:\Users\dario\Documents\B_all_coluna_', num2str(k),
'de10.txt'];
    file_ID=fopen(nome_out, 'w');
    fprintf(file_ID,'%d %d %d %d\n', B_subset);
    fclose(file_ID);
end

%% exportar indices dos pixeis de area continental
%indices de NaN
ind_NaN=find((B_all_coluna(:,1)<= 0) & (B_all_coluna(:,2)<=0) &
(B_all_coluna(:,3)<= 0) & (B_all_coluna(:,4)<= 0));
[lnh, cln]=size(B1);

dim_NaN=size(ind_NaN);
ind_NaN(dim_NaN(1)+1)=lnh;ind_NaN(dim_NaN(1)+2)=cln;
% ind_NaN(end+1)=lnh;ind_NaN(end+2)=cln;
[nr_linhas_NaN,nr_colunas_NaN]=size(ind_NaN);
nr_lnh = round(nr_linhas_NaN/10);
for k = 1:10
    if k < 10
```

## Árvores de Decisão Aplicadas à Detecção de Formas Costeiras Através de Imagens IKONOS-2

### Anexos

```
        ind_NaN_div = ind_NaN((1+(k-1)*nr_lrhs):nr_lrhs*k,:);  
    else  
        ind_NaN_div= ind_NaN((1+(k-1)*nr_lrhs):end,:);  
    end  
  
    nome_out = ['C:\Users\dario\Documents\ind_NaN', num2str(k),  
'de10.txt'];  
    file_ID=fopen(nome_out, 'w');  
    fprintf(file_ID, '%d\n', ind_NaN_div);  
    fclose(file_ID);  
end
```

Secção B

##CAP1 - Classificação

#aprendizagem

rm(list=ls())

nr\_treino=c(4133,4091,329,88,1001)

# Dados de treino

dados\_treino = read.table("dados\_isprs\_treino.txt", sep="\t")

dim\_dados\_treino=dim(dados\_treino)

# dados de teste

dados\_teste = read.table("dados\_isprs\_teste.txt", sep="\t")

dim\_dados\_teste=dim(dados\_teste)

# arvore de decisao com rpart

library(rpart)

dados=data.frame(dados\_treino[,1:4], classes = factor(c(rep("1",nr\_treino[1]),  
rep("2", nr\_treino[2]), rep("3", nr\_treino[3]), rep("4", nr\_treino[4]), rep("5",  
nr\_treino[5]))))

arvore=rpart(classes~., data=dados,cp=0,xval=0)

#ilustrar a arvore

dev.new(1)

plot(arvore,uniform=TRUE, main="Árvore - classificação praias")

text(arvore,digits=0,font=6,fheight=0.8,cex=.8)

dev.copy(jpeg,"arvore\_prais.jpeg")

# arvore de decisao com rpart (Poda, com cp=0.01 e numero de cv=5)

arvore\_prune=rpart(classes~., data=dados,cp=0.01,xval=5)

#ilustrar a arvore Podada

```
dev.new(2)

plot(arvore_prune,uniform=TRUE, main="Árvore Podada - classificação
praias")

text(arvore_prune,digits=0,fancy=TRUE,font=6,fheight=0.8,cex=.8)

dev.copy(jpeg,"arvore_podada_praias.jpeg")


#resumo dos erros das divisões
CP=printcp(arvore_prune)

write.table(CP, "CP_praias.txt",sep="\t",quote=FALSE, row.names=FALSE,
col.names=FALSE)


#ilustar os erros
dev.new(3)

plotcp(arvore_prune)

dev.copy(jpeg,"erro_relativo_praia.jpeg")

dev.off()


#sumario da arvore
sumario=summary(arvore_prune)

#write.table(sumario,
paste("sumario_tree_",ano[k],".txt",sep=""),sep="\t",quote=FALSE,
row.names=FALSE, col.names=FALSE)


# matriz de perdas
L=residuals(arvore_prune)

write.table(L, "matrix_perdas.txt",sep="\t",quote=FALSE, row.names=FALSE,
col.names=FALSE)


# previsao
previsao = predict(arvore, dados_teste[,1:4], type="class")
```

```
# resultados

mc=table(dados_teste[,5], previsao)#matriz de confusão

#print(mc)


# export tree

save(arvore, file=paste("arvore_R_praias", ".RData", sep=""))

save(arvore, file="arvore_R_praias.txt", ascii=TRUE)


#precisão e matriz de confusão

i=1 #indice

y=1 #indice


pG=0 #variavel para armazenar a precisão global

pK=0 #variavel para armazenar o parametro estatistico kappa

Tmc=sum(mc)      #nr total de pixels

lim=dim(mc) #dimensao da matrix de confusao

pUtilizador=matrix(1,lim[1],1) # vetor unitario de armazenamento das precões de
utilizador

pProdutor=matrix(1,1,lim[1]) # vetor unitario de armazenamento das precões de
produtor


for (i in seq(lim[1]))
{
  pU=0
  pP=0
  for (y in seq(lim[1]))
  {
    pU= pU+mc[i,y]
    pP= pP+mc[y,i]
  }
}
```

```
pUtilizador[i]=(mc[i,i]/pU)*100 #precisao do utilizador
pProdutor[i]=(mc[i,i]/pP)*100      #precisao do produtor
pG=pG+mc[i,i]
pK=pK+(pU/Tmc)*(pP/Tmc)
}

pKappa=((pG/Tmc)-pK)/(1-pK) #parametro estatistico kappa
pGlobal=pG/sum(mc)*100 #precisão Global

##leitura e predict da imagem

for (p in seq(10))
{
  documento=paste("B_all_coluna_",toString(p),"de10.txt", sep="")
  Linha_p_10=read.table(documento, header=FALSE, sep="")
  img_prev=predict(arvore_prune,Linha_p_10,type=c("vector"))
  write.table(img_prev,
paste("previsao_",toString(p),"de10.txt"),sep="\t",quote=FALSE, row.names=FALSE,
col.names=FALSE)
}

write.table(pKappa, "kappa_praias.txt" ,sep="\t",quote=FALSE, row.names=FALSE,
col.names=FALSE)

write.table(pGlobal, "OA_praias.txt" ,sep="\t",quote=FALSE, row.names=FALSE,
col.names=FALSE)

write.table(pUtilizador, "precisao_utilizador_praias.txt" ,sep="\t",quote=FALSE,
row.names=FALSE, col.names=FALSE)

write.table(pProdutor, "precisao_produtores_praias.txt" ,sep="\t",quote=FALSE,
row.names=FALSE, col.names=FALSE)
```

Seção C

Call:

```
rpart(formula = classes ~ ., data = dados, cp = 0, xval = 0)
```

```
n= 9642
```

	CP	nsplit	rel error
1	6.652750e-01	0	1.000000000
2	1.817027e-01	1	0.33472500
3	4.538029e-02	2	0.15302233
4	1.343256e-02	3	0.10764204
5	1.252496e-02	4	0.09420948
6	1.043747e-02	5	0.08168452
7	6.353240e-03	7	0.06080958
8	4.810310e-03	9	0.04810310
9	2.178254e-03	11	0.03848248
10	9.076057e-04	12	0.03630423
11	6.655775e-04	16	0.03267381
12	3.630423e-04	19	0.03067707
13	1.815211e-04	22	0.02958795
14	9.076057e-05	23	0.02940643
15	6.050705e-05	25	0.02922490
16	0.000000e+00	31	0.02886186

Variable importance

V2 V3 V1 V4

31 27 25 17

Node number 1: 9642 observations, complexity param=0.665275

predicted class=1 expected loss=0.5713545 P(node) =1

class counts: 4133 4091 329 88 1001

probabilities: 0.429 0.424 0.034 0.009 0.104

left son=2 (3965 obs) right son=3 (5677 obs)

Primary splits:

V2 < 26.5 to the left, improve=3069.649, (0 missing)

V3 < 30.5 to the left, improve=2097.214, (0 missing)

V1 < 11.5 to the left, improve=1789.778, (0 missing)

V4 < 23.5 to the left, improve=1274.249, (0 missing)

Surrogate splits:

V3 < 30.5 to the left, agree=0.917, adj=0.797, (0 split)

V1 < 11.5 to the left, agree=0.885, adj=0.721, (0 split)

V4 < 7.5 to the left, agree=0.708, adj=0.289, (0 split)

Node number 2: 3965 observations, complexity param=0.00635324

predicted class=1 expected loss=0.03253468 P(node) =0.4112217

class counts: 3836 129 0 0 0

probabilities: 0.967 0.033 0.000 0.000 0.000

left son=4 (3549 obs) right son=5 (416 obs)

Primary splits:

V2 < 25.5 to the left, improve=48.951660, (0 missing)

V3 < 29.5 to the left, improve=11.950630, (0 missing)

V1 < 10.5 to the left, improve= 4.692176, (0 missing)

V4 < 7.5 to the right, improve= 1.779143, (0 missing)

Surrogate splits:

V3 < 30.5 to the left, agree=0.911, adj=0.154, (0 split)

V4 < 9.5 to the left, agree=0.907, adj=0.115, (0 split)

V1 < 12.5 to the left, agree=0.904, adj=0.084, (0 split)

Node number 3: 5677 observations, complexity param=0.1817027

predicted class=2 expected loss=0.3020962 P(node) =0.5887783

class counts: 297 3962 329 88 1001

probabilities: 0.052 0.698 0.058 0.016 0.176

left son=6 (4273 obs) right son=7 (1404 obs)



Primary splits:

V1 < 31.5 to the left, improve=1511.062, (0 missing)

V4 < 21.5 to the left, improve=1510.911, (0 missing)

V2 < 47.5 to the left, improve=1506.481, (0 missing)

V3 < 41.5 to the left, improve=1496.189, (0 missing)

Surrogate splits:

V2 < 43.5 to the left, agree=0.999, adj=0.995, (0 split)

V4 < 20.5 to the left, agree=0.996, adj=0.986, (0 split)

V3 < 41.5 to the left, agree=0.996, adj=0.983, (0 split)

Node number 4: 3549 observations

predicted class=1 expected loss=0.00563539 P(node) =0.3680772

class counts: 3529 20 0 0 0

probabilities: 0.994 0.006 0.000 0.000 0.000

Node number 5: 416 observations, complexity param=0.00635324

predicted class=1 expected loss=0.2620192 P(node) =0.04314458

class counts: 307 109 0 0 0

probabilities: 0.738 0.262 0.000 0.000 0.000

left son=10 (302 obs) right son=11 (114 obs)

Primary splits:

V4 < 7.5 to the right, improve=93.284940, (0 missing)

V1 < 11.5 to the right, improve=47.885970, (0 missing)

V3 < 30.5 to the right, improve= 8.421302, (0 missing)

Surrogate splits:

V1 < 11.5 to the right, agree=0.868, adj=0.518, (0 split)

V3 < 28.5 to the right, agree=0.731, adj=0.018, (0 split)

Node number 6: 4273 observations, complexity param=0.01043747

predicted class=2 expected loss=0.07278259 P(node) =0.4431653

class counts: 297 3962 1 13 0

probabilities: 0.070 0.927 0.000 0.003 0.000

left son=12 (375 obs) right son=13 (3898 obs)

Primary splits:

V2 < 27.5 to the left, improve=76.414360, (0 missing)

V4 < 9.5 to the right, improve=49.537610, (0 missing)

V1 < 26.5 to the left, improve=14.818990, (0 missing)

V3 < 31.5 to the left, improve= 3.124979, (0 missing)

Surrogate splits:

V1 < 10.5 to the left, agree=0.913, adj=0.011, (0 split)

Node number 7: 1404 observations, complexity param=0.04538029

predicted class=5 expected loss=0.287037 P(node) =0.1456129

class counts: 0 0 328 75 1001

probabilities: 0.000 0.000 0.234 0.053 0.713

left son=14 (256 obs) right son=15 (1148 obs)

Primary splits:

V3 < 72.5 to the right, improve=333.1602, (0 missing)

V4 < 77.5 to the left, improve=310.1496, (0 missing)

V2 < 104.5 to the right, improve=220.6350, (0 missing)

V1 < 78.5 to the left, improve=155.5623, (0 missing)

Surrogate splits:

V2 < 104.5 to the right, agree=0.947, adj=0.707, (0 split)

V1 < 104.5 to the right, agree=0.888, adj=0.387, (0 split)

V4 < 101.5 to the right, agree=0.826, adj=0.047, (0 split)

Node number 10: 302 observations

predicted class=1 expected loss=0.05629139 P(node) =0.0313213

class counts: 285 17 0 0 0

probabilities: 0.944 0.056 0.000 0.000 0.000

Node number 11: 114 observations, complexity param=9.076057e-05

predicted class=2 expected loss=0.1929825 P(node) =0.01182327

class counts: 22 92 0 0 0

probabilities: 0.193 0.807 0.000 0.000 0.000

left son=22 (70 obs) right son=23 (44 obs)

Primary splits:

V4 < 6.5 to the right, improve=3.119162, (0 missing)

V3 < 29.5 to the left, improve=1.819064, (0 missing)

V1 < 11.5 to the right, improve=1.224719, (0 missing)

Node number 12: 375 observations, complexity param=0.01043747

predicted class=2 expected loss=0.376 P(node) =0.03889235

class counts: 141 234 0 0 0

probabilities: 0.376 0.624 0.000 0.000 0.000

left son=24 (139 obs) right son=25 (236 obs)

Primary splits:

V4 < 8.5 to the right, improve=127.70100, (0 missing)

V1 < 12.5 to the right, improve= 66.78924, (0 missing)

V3 < 30.5 to the right, improve= 8.09222, (0 missing)

Surrogate splits:

V1 < 12.5 to the right, agree=0.816, adj=0.504, (0 split)

Node number 13: 3898 observations, complexity param=0.00481031

predicted class=2 expected loss=0.04361211 P(node) =0.404273

class counts: 156 3728 1 13 0

probabilities: 0.040 0.956 0.000 0.003 0.000

left son=26 (736 obs) right son=27 (3162 obs)

Primary splits:

V4 < 10.5 to the right, improve=30.6704700, (0 missing)

V1 < 26.5 to the left, improve=15.4539600, (0 missing)

V2 < 28.5 to the left, improve= 5.6213040, (0 missing)

V3 < 35.5 to the left, improve= 0.5336586, (0 missing)

Surrogate splits:

V1 < 16.5 to the right, agree=0.901, adj=0.478, (0 split)

V3 < 34.5 to the right, agree=0.891, adj=0.423, (0 split)

V2 < 32.5 to the right, agree=0.880, adj=0.364, (0 split)

Node number 14: 256 observations

predicted class=3 expected loss=0.01171875 P(node) =0.02655051

class counts: 0 0 253 0 3

probabilities: 0.000 0.000 0.988 0.000 0.012

Node number 15: 1148 observations, complexity param=0.01343256

predicted class=5 expected loss=0.130662 P(node) =0.1190624

class counts: 0 0 75 75 998

probabilities: 0.000 0.000 0.065 0.065 0.869

left son=30 (151 obs) right son=31 (997 obs)

Primary splits:

V4 < 71 to the left, improve=194.1110, (0 missing)

V1 < 71 to the left, improve=191.1325, (0 missing)

V2 < 72.5 to the left, improve=139.3426, (0 missing)

V3 < 53.5 to the left, improve=128.6527, (0 missing)

Surrogate splits:

V1 < 71 to the left, agree=0.997, adj=0.974, (0 split)

V2 < 76.5 to the left, agree=0.961, adj=0.702, (0 split)

V3 < 53.5 to the left, agree=0.944, adj=0.576, (0 split)

Node number 22: 70 observations, complexity param=9.076057e-05

predicted class=2 expected loss=0.2857143 P(node) =0.007259905

class counts: 20 50 0 0 0

probabilities: 0.286 0.714 0.000 0.000 0.000

left son=44 (13 obs) right son=45 (57 obs)

Primary splits:

V3 < 29.5 to the left, improve=2.0397150, (0 missing)

V1 < 10.5 to the left, improve=0.3174603, (0 missing)

Surrogate splits:

V1 < 10.5 to the left, agree=0.857, adj=0.231, (0 split)

Node number 23: 44 observations

predicted class=2 expected loss=0.04545455 P(node) =0.004563369

class counts: 2 42 0 0 0

probabilities: 0.045 0.955 0.000 0.000 0.000

Node number 24: 139 observations, complexity param=6.050705e-05

predicted class=1 expected loss=0.08633094 P(node) =0.0144161

class counts: 127 12 0 0 0

probabilities: 0.914 0.086 0.000 0.000 0.000

left son=48 (87 obs) right son=49 (52 obs)

Primary splits:

V4 < 9.5 to the right, improve=2.6048920, (0 missing)

V1 < 12.5 to the right, improve=1.1395180, (0 missing)

V3 < 30.5 to the left, improve=0.4324054, (0 missing)

Surrogate splits:

V3 < 30.5 to the right, agree=0.669, adj=0.115, (0 split)

V1 < 12.5 to the right, agree=0.647, adj=0.058, (0 split)

Node number 25: 236 observations, complexity param=6.050705e-05

predicted class=2 expected loss=0.05932203 P(node) =0.02447625

class counts: 14 222 0 0 0

probabilities: 0.059 0.941 0.000 0.000 0.000

left son=50 (59 obs) right son=51 (177 obs)

Primary splits:

V4 < 7.5 to the right, improve=3.2655370, (0 missing)

V1 < 12.5 to the right, improve=0.9019263, (0 missing)

V3 < 29.5 to the right, improve=0.1208012, (0 missing)

Surrogate splits:

V1 < 12.5 to the right, agree=0.801, adj=0.203, (0 split)

Node number 26: 736 observations, complexity param=0.00481031

predicted class=2 expected loss=0.1807065 P(node) =0.07633271

class counts: 119 603 1 13 0

probabilities: 0.162 0.819 0.001 0.018 0.000

left son=52 (85 obs) right son=53 (651 obs)

Primary splits:

V2 < 29.5 to the left, improve=78.91134, (0 missing)

V3 < 32.5 to the left, improve=47.71859, (0 missing)

V4 < 24.5 to the left, improve=20.19698, (0 missing)

V1 < 15.5 to the left, improve=14.25310, (0 missing)

Surrogate splits:

V3 < 32.5 to the left, agree=0.923, adj=0.329, (0 split)

V1 < 14.5 to the left, agree=0.901, adj=0.141, (0 split)

Node number 27: 3162 observations, complexity param=0.0006655775

predicted class=2 expected loss=0.01170145 P(node) =0.3279403

class counts: 37 3125 0 0 0

probabilities: 0.012 0.988 0.000 0.000 0.000

left son=54 (411 obs) right son=55 (2751 obs)

Primary splits:

V4 < 9.5 to the right, improve=1.8507950, (0 missing)

V2 < 28.5 to the left, improve=1.4086120, (0 missing)

V3 < 32.5 to the left, improve=0.2822032, (0 missing)

V1 < 12.5 to the right, improve=0.0802317, (0 missing)

Surrogate splits:

V3 < 33.5 to the right, agree=0.882, adj=0.095, (0 split)

V1 < 15.5 to the right, agree=0.877, adj=0.054, (0 split)

V2 < 32.5 to the right, agree=0.873, adj=0.022, (0 split)

Node number 30: 151 observations, complexity param=0.01252496

predicted class=3 expected loss=0.5033113 P(node) =0.01566065

class counts: 0 0 75 75 1

probabilities: 0.000 0.000 0.497 0.497 0.007

left son=60 (72 obs) right son=61 (79 obs)

Primary splits:

V3 < 48.5 to the right, improve=63.20631, (0 missing)

V2 < 63.5 to the right, improve=47.64302, (0 missing)

V1 < 60.5 to the right, improve=16.76938, (0 missing)

V4 < 46.5 to the left, improve=16.30116, (0 missing)

Surrogate splits:

V2 < 61.5 to the right, agree=0.921, adj=0.833, (0 split)

V1 < 57.5 to the right, agree=0.742, adj=0.458, (0 split)

V4 < 44.5 to the left, agree=0.689, adj=0.347, (0 split)

Node number 31: 997 observations

predicted class=5 expected loss=0 P(node) =0.1034018

class counts: 0 0 0 0 997

probabilities: 0.000 0.000 0.000 0.000 1.000

Node number 44: 13 observations

predicted class=1 expected loss=0.4615385 P(node) =0.001348268

class counts: 7 6 0 0 0

probabilities: 0.538 0.462 0.000 0.000 0.000

Node number 45: 57 observations

predicted class=2 expected loss=0.2280702 P(node) =0.005911637

class counts: 13 44 0 0 0

probabilities: 0.228 0.772 0.000 0.000 0.000

Node number 48: 87 observations

predicted class=1 expected loss=0.01149425 P(node) =0.009023024

class counts: 86 1 0 0 0

probabilities: 0.989 0.011 0.000 0.000 0.000

Node number 49: 52 observations, complexity param=6.050705e-05

predicted class=1 expected loss=0.2115385 P(node) =0.005393072

class counts: 41 11 0 0 0

probabilities: 0.788 0.212 0.000 0.000 0.000

left son=98 (15 obs) right son=99 (37 obs)

Primary splits:

V3 < 30.5 to the left, improve=1.886694, (0 missing)

V1 < 12.5 to the right, improve=1.312821, (0 missing)

Surrogate splits:

V1 < 14.5 to the right, agree=0.75, adj=0.133, (0 split)

Node number 50: 59 observations, complexity param=6.050705e-05

predicted class=2 expected loss=0.2033898 P(node) =0.006119062

class counts: 12 47 0 0 0

probabilities: 0.203 0.797 0.000 0.000 0.000

left son=100 (35 obs) right son=101 (24 obs)

Primary splits:

V3 < 31.5 to the left, improve=2.1162630, (0 missing)

V1 < 12.5 to the right, improve=0.1094487, (0 missing)

Surrogate splits:

V1 < 12.5 to the left, agree=0.661, adj=0.167, (0 split)

Node number 51: 177 observations

predicted class=2 expected loss=0.01129944 P(node) =0.01835719

class counts: 2 175 0 0 0



probabilities: 0.011 0.989 0.000 0.000 0.000

Node number 52: 85 observations, complexity param=0.0003630423

predicted class=1 expected loss=0.1882353 P(node) =0.008815598

class counts: 69 16 0 0 0

probabilities: 0.812 0.188 0.000 0.000 0.000

left son=104 (47 obs) right son=105 (38 obs)

Primary splits:

V2 < 28.5 to the left, improve=4.462473, (0 missing)

V3 < 32.5 to the left, improve=3.486606, (0 missing)

V4 < 11.5 to the right, improve=2.045042, (0 missing)

V1 < 14.5 to the right, improve=1.408165, (0 missing)

Surrogate splits:

V3 < 32.5 to the left, agree=0.706, adj=0.342, (0 split)

V4 < 12.5 to the left, agree=0.588, adj=0.079, (0 split)

V1 < 17.5 to the left, agree=0.576, adj=0.053, (0 split)

Node number 53: 651 observations, complexity param=0.002178254

predicted class=2 expected loss=0.09831029 P(node) =0.06751711

class counts: 50 587 1 13 0

probabilities: 0.077 0.902 0.002 0.020 0.000

left son=106 (637 obs) right son=107 (14 obs)

Primary splits:

V4 < 24.5 to the left, improve=21.759040, (0 missing)

V1 < 26.5 to the left, improve=14.205860, (0 missing)

V2 < 30.5 to the left, improve= 6.434423, (0 missing)

V3 < 33.5 to the left, improve= 3.620192, (0 missing)

Surrogate splits:

V1 < 26.5 to the left, agree=0.989, adj=0.5, (0 split)

Node number 54: 411 observations, complexity param=0.0006655775

predicted class=2 expected loss=0.05596107 P(node) =0.04262601

class counts: 23 388 0 0 0

probabilities: 0.056 0.944 0.000 0.000 0.000

left son=108 (32 obs) right son=109 (379 obs)

Primary splits:

V2 < 28.5 to the left, improve=15.678260, (0 missing)

V3 < 31.5 to the left, improve=14.837300, (0 missing)

V1 < 14.5 to the left, improve= 3.416934, (0 missing)

Surrogate splits:

V3 < 30.5 to the left, agree=0.927, adj=0.062, (0 split)

Node number 55: 2751 observations

predicted class=2 expected loss=0.005089059 P(node) =0.2853143

class counts: 14 2737 0 0 0

probabilities: 0.005 0.995 0.000 0.000 0.000

Node number 60: 72 observations

predicted class=3 expected loss=0.02777778 P(node) =0.00746733

class counts: 0 0 70 1 1

probabilities: 0.000 0.000 0.972 0.014 0.014

Node number 61: 79 observations

predicted class=4 expected loss=0.06329114 P(node) =0.008193321

class counts: 0 0 5 74 0

probabilities: 0.000 0.000 0.063 0.937 0.000

Node number 98: 15 observations

predicted class=1 expected loss=0 P(node) =0.001555694

class counts: 15 0 0 0 0

probabilities: 1.000 0.000 0.000 0.000 0.000

Node number 99: 37 observations, complexity param=6.050705e-05

predicted class=1 expected loss=0.2972973 P(node) =0.003837378

class counts: 26 11 0 0 0

probabilities: 0.703 0.297 0.000 0.000 0.000

left son=198 (28 obs) right son=199 (9 obs)

Primary splits:

V1 < 12.5 to the right, improve=1.586444, (0 missing)

V3 < 31.5 to the left, improve=0.515015, (0 missing)

Node number 100: 35 observations, complexity param=6.050705e-05

predicted class=2 expected loss=0.3142857 P(node) =0.003629952

class counts: 11 24 0 0 0

probabilities: 0.314 0.686 0.000 0.000 0.000

left son=200 (13 obs) right son=201 (22 obs)

Primary splits:

V1 < 12.5 to the right, improve=2.0787210, (0 missing)

V3 < 30.5 to the left, improve=0.3857143, (0 missing)

Surrogate splits:

V3 < 30.5 to the left, agree=0.657, adj=0.077, (0 split)

Node number 101: 24 observations

predicted class=2 expected loss=0.04166667 P(node) =0.00248911

class counts: 1 23 0 0 0

probabilities: 0.042 0.958 0.000 0.000 0.000

Node number 104: 47 observations

predicted class=1 expected loss=0.04255319 P(node) =0.004874507

class counts: 45 2 0 0 0

probabilities: 0.957 0.043 0.000 0.000 0.000

Node number 105: 38 observations, complexity param=0.0003630423

predicted class=1 expected loss=0.3684211 P(node) =0.003941091

class counts: 24 14 0 0 0

probabilities: 0.632 0.368 0.000 0.000 0.000

left son=210 (17 obs) right son=211 (21 obs)

Primary splits:

V4 < 11.5 to the right, improve=3.869084, (0 missing)

V3 < 32.5 to the left, improve=1.819595, (0 missing)

V1 < 14.5 to the right, improve=1.455639, (0 missing)

Surrogate splits:

V1 < 15.5 to the right, agree=0.658, adj=0.235, (0 split)

V3 < 33.5 to the right, agree=0.579, adj=0.059, (0 split)

Node number 106: 637 observations, complexity param=0.0009076057

predicted class=2 expected loss=0.08006279 P(node) =0.06606513

class counts: 50 586 1 0 0

probabilities: 0.078 0.920 0.002 0.000 0.000

left son=212 (63 obs) right son=213 (574 obs)

Primary splits:

V2 < 30.5 to the left, improve=6.9108290, (0 missing)

V4 < 12.5 to the right, improve=5.8228860, (0 missing)

V3 < 33.5 to the left, improve=4.0909640, (0 missing)

V1 < 18.5 to the right, improve=0.4869106, (0 missing)

Node number 107: 14 observations

predicted class=4 expected loss=0.07142857 P(node) =0.001451981

class counts: 0 1 0 13 0

probabilities: 0.000 0.071 0.000 0.929 0.000

Node number 108: 32 observations, complexity param=0.0006655775

predicted class=1 expected loss=0.46875 P(node) =0.003318814

class counts: 17 15 0 0 0

probabilities: 0.531 0.469 0.000 0.000 0.000

left son=216 (11 obs) right son=217 (21 obs)

Primary splits:

V3 < 31.5 to the left, improve=7.3660710, (0 missing)

V1 < 13.5 to the right, improve=0.1887077, (0 missing)

Node number 109: 379 observations

predicted class=2 expected loss=0.01583113 P(node) =0.0393072

class counts: 6 373 0 0 0

probabilities: 0.016 0.984 0.000 0.000 0.000

Node number 198: 28 observations

predicted class=1 expected loss=0.2142857 P(node) =0.002903962

class counts: 22 6 0 0 0

probabilities: 0.786 0.214 0.000 0.000 0.000

Node number 199: 9 observations

predicted class=2 expected loss=0.4444444 P(node) =0.0009334163

class counts: 4 5 0 0 0

probabilities: 0.444 0.556 0.000 0.000 0.000

Node number 200: 13 observations

predicted class=1 expected loss=0.4615385 P(node) =0.001348268

class counts: 7 6 0 0 0

probabilities: 0.538 0.462 0.000 0.000 0.000

Node number 201: 22 observations

predicted class=2 expected loss=0.1818182 P(node) =0.002281684

class counts: 4 18 0 0 0

probabilities: 0.182 0.818 0.000 0.000 0.000

Node number 210: 17 observations

predicted class=1 expected loss=0.1176471 P(node) =0.00176312

class counts: 15 2 0 0 0

probabilities: 0.882 0.118 0.000 0.000 0.000

Node number 211: 21 observations, complexity param=0.0003630423

predicted class=2 expected loss=0.4285714 P(node) =0.002177971

class counts: 9 12 0 0 0

probabilities: 0.429 0.571 0.000 0.000 0.000

left son=422 (7 obs) right son=423 (14 obs)

Primary splits:

V3 < 32.5 to the left, improve=1.7142860, (0 missing)

V1 < 14.5 to the right, improve=0.8241758, (0 missing)

Node number 212: 63 observations, complexity param=0.0009076057

predicted class=2 expected loss=0.3015873 P(node) =0.006533914

class counts: 19 44 0 0 0

probabilities: 0.302 0.698 0.000 0.000 0.000

left son=424 (27 obs) right son=425 (36 obs)

Primary splits:

V4 < 11.5 to the right, improve=10.169310, (0 missing)

V1 < 16.5 to the right, improve= 3.951447, (0 missing)

V3 < 33.5 to the left, improve= 2.747344, (0 missing)

Surrogate splits:

V1 < 16.5 to the right, agree=0.730, adj=0.370, (0 split)

V3 < 32.5 to the left, agree=0.603, adj=0.074, (0 split)

Node number 213: 574 observations, complexity param=0.0009076057

predicted class=2 expected loss=0.05574913 P(node) =0.05953122

class counts: 31 542 1 0 0

probabilities: 0.054 0.944 0.002 0.000 0.000

left son=426 (134 obs) right son=427 (440 obs)

Primary splits:

V4 < 13.5 to the right, improve=6.4201240, (0 missing)

V1 < 18.5 to the right, improve=1.4604160, (0 missing)

V2 < 33.5 to the left, improve=0.8071398, (0 missing)

V3 < 33.5 to the left, improve=0.5380289, (0 missing)

Surrogate splits:

V1 < 19.5 to the right, agree=0.876, adj=0.470, (0 split)

V2 < 35.5 to the right, agree=0.861, adj=0.403, (0 split)

V3 < 36.5 to the right, agree=0.854, adj=0.373, (0 split)

Node number 216: 11 observations

predicted class=1 expected loss=0 P(node) =0.001140842

class counts: 11 0 0 0 0

probabilities: 1.000 0.000 0.000 0.000 0.000

Node number 217: 21 observations

predicted class=2 expected loss=0.2857143 P(node) =0.002177971

class counts: 6 15 0 0 0

probabilities: 0.286 0.714 0.000 0.000 0.000

Node number 422: 7 observations

predicted class=1 expected loss=0.2857143 P(node) =0.0007259905

class counts: 5 2 0 0 0

probabilities: 0.714 0.286 0.000 0.000 0.000

Node number 423: 14 observations

predicted class=2 expected loss=0.2857143 P(node) =0.001451981

class counts: 4 10 0 0 0

probabilities: 0.286 0.714 0.000 0.000 0.000

Node number 424: 27 observations, complexity param=0.0001815211

predicted class=1 expected loss=0.3703704 P(node) =0.002800249

class counts: 17 10 0 0 0

probabilities: 0.630 0.370 0.000 0.000 0.000

left son=848 (16 obs) right son=849 (11 obs)

Primary splits:

V3 < 33.5 to the left, improve=1.1380470, (0 missing)

V1 < 15.5 to the right, improve=0.7640212, (0 missing)

V4 < 12.5 to the right, improve=0.6259259, (0 missing)

Surrogate splits:

V1 < 17.5 to the left, agree=0.667, adj=0.182, (0 split)

Node number 425: 36 observations

predicted class=2 expected loss=0.05555556 P(node) =0.003733665

class counts: 2 34 0 0 0

probabilities: 0.056 0.944 0.000 0.000 0.000

Node number 426: 134 observations, complexity param=0.0009076057

predicted class=2 expected loss=0.1940299 P(node) =0.01389753

class counts: 25 108 1 0 0

probabilities: 0.187 0.806 0.007 0.000 0.000

left son=852 (29 obs) right son=853 (105 obs)

Primary splits:

V2 < 33.5 to the left, improve=21.097380, (0 missing)

V3 < 34.5 to the left, improve= 8.984853, (0 missing)

V1 < 16.5 to the left, improve= 6.660471, (0 missing)

V4 < 17.5 to the left, improve= 1.106634, (0 missing)

Surrogate splits:

V3 < 34.5 to the left, agree=0.858, adj=0.345, (0 split)

V1 < 16.5 to the left, agree=0.851, adj=0.310, (0 split)



Node number 427: 440 observations

predicted class=2 expected loss=0.01363636 P(node) =0.04563369

class counts: 6 434 0 0 0

probabilities: 0.014 0.986 0.000 0.000 0.000

Node number 848: 16 observations

predicted class=1 expected loss=0.25 P(node) =0.001659407

class counts: 12 4 0 0 0

probabilities: 0.750 0.250 0.000 0.000 0.000

Node number 849: 11 observations

predicted class=2 expected loss=0.4545455 P(node) =0.001140842

class counts: 5 6 0 0 0

probabilities: 0.455 0.545 0.000 0.000 0.000

Node number 852: 29 observations

predicted class=1 expected loss=0.2758621 P(node) =0.003007675

class counts: 21 8 0 0 0

probabilities: 0.724 0.276 0.000 0.000 0.000

Node number 853: 105 observations

predicted class=2 expected loss=0.04761905 P(node) =0.01088986

class counts: 4 100 1 0 0

probabilities: 0.038 0.952 0.010 0.000 0.000

## Secção D

```
% ilustrar a imagem classificada
nova_pasta='C:\Users\dario\Documents\';

%carregar indices da área continental
for i=1:10
    nome=[nova_pasta,'ind_NaN', num2str(i), 'de10.txt'];
    P_ind= textread(nome);
    if i > 5
        if i == 6
            P_ind2=P_ind';
        else
            P_ind2=[P_ind2 P_ind'];
        end
    else
        if i == 1
            P_ind1=P_ind';
        else
            P_ind1=[P_ind1 P_ind'];
        end
    end
end

ind_NaN_linha=[P_ind1 P_ind2];
ind_NaN=ind_NaN_linha';
%extração da dimensão das imagens
img_nr_lnh=ind_NaN(end-1);img_nr_cln=ind_NaN(end);

%% ilustrar a imagem classificada
%Leitura das classes da função predict

nome_1=[nova_pasta,'previsao_ 1 de10.txt'];
nome_2=[nova_pasta,'previsao_ 2 de10.txt'];
nome_3=[nova_pasta,'previsao_ 3 de10.txt'];
nome_4=[nova_pasta,'previsao_ 4 de10.txt'];
nome_5=[nova_pasta,'previsao_ 5 de10.txt'];
nome_6=[nova_pasta,'previsao_ 6 de10.txt'];
nome_7=[nova_pasta,'previsao_ 7 de10.txt'];
nome_8=[nova_pasta,'previsao_ 8 de10.txt'];
nome_9=[nova_pasta,'previsao_ 9 de10.txt'];
nome_10=[nova_pasta,'previsao_ 10 de10.txt'];

P1= textread(nome_1);
P2= textread(nome_2);
P3= textread(nome_3);
P4= textread(nome_4);
P5= textread(nome_5);
P6= textread(nome_6);
P7= textread(nome_7);
P8= textread(nome_8);
P9= textread(nome_9);
P10=textread(nome_10);

P_linha=[P1' P2' P3' P4' P5' P6' P7' P8' P9' P10']; %vetor linha com
total de "pixeis"(classes)
P_coluna=P_linha'; %vetor transposta do P_linha
```

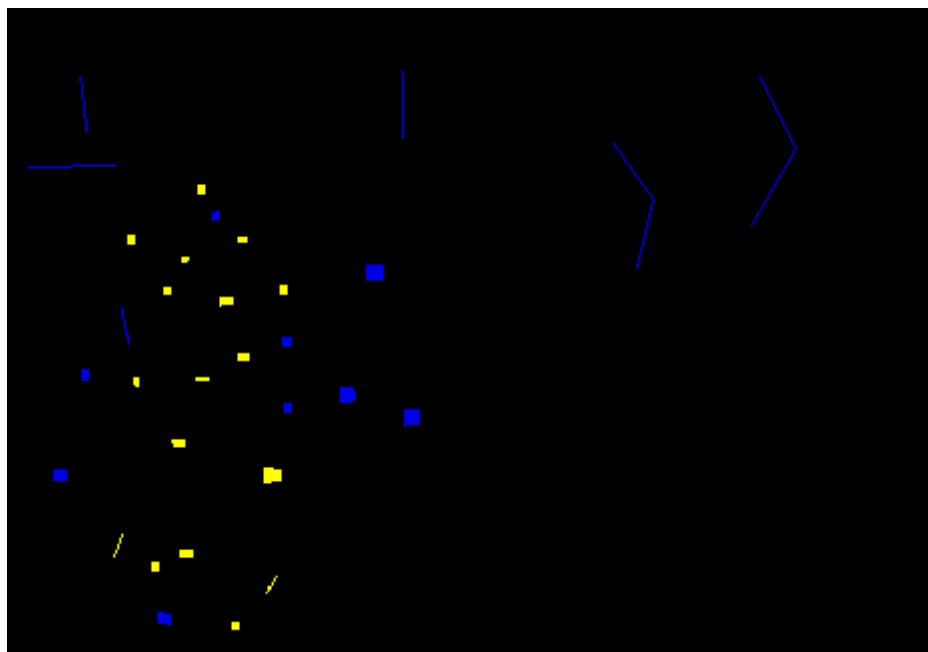
```
% passar de vetor classes para matrix
img_matrix=reshape(P_coluna,img_nr_lnh,img_nr_cln); %transforma o
vetor em matrix
img_matrix(ind_NaN(1:end-2))=0;

%% Representação da imagem
X_aux1 = double(img_matrix);
X_aux2 = uint8((X_aux1-min(X_aux1(:)))/(max(X_aux1(:))-
min(X_aux1(:)))*255);
[X,map]=gray2ind(X_aux2,6);
mapa_cores = [0 0 0 % terra
              25 25 112 % mar
              184 134 11 % sedimentos
              255 255 255 % rebentação
              238 221 130 % areia húmida
              255 255 0]; % areia seca
mapa_cores = mapa_cores./255;
faixa_esq = 40; % para eliminar uma faixa que aparece do lado
esquerdo...
figure
imshow(ind2rgb(X(:,faixa_esq:end), mapa_cores), 'border', 'tight')
```

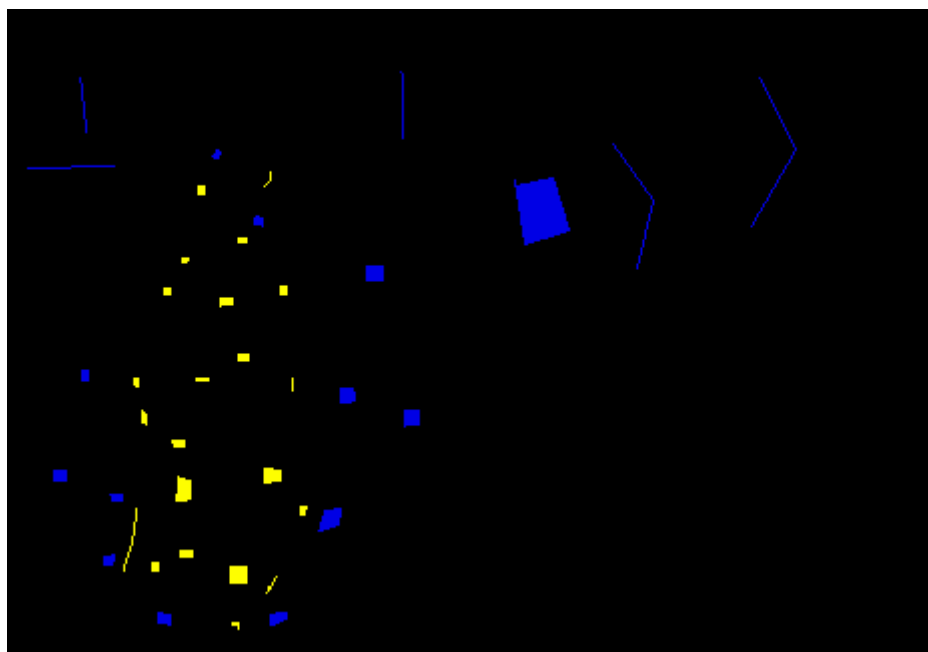
## Anexo II

### Secção A

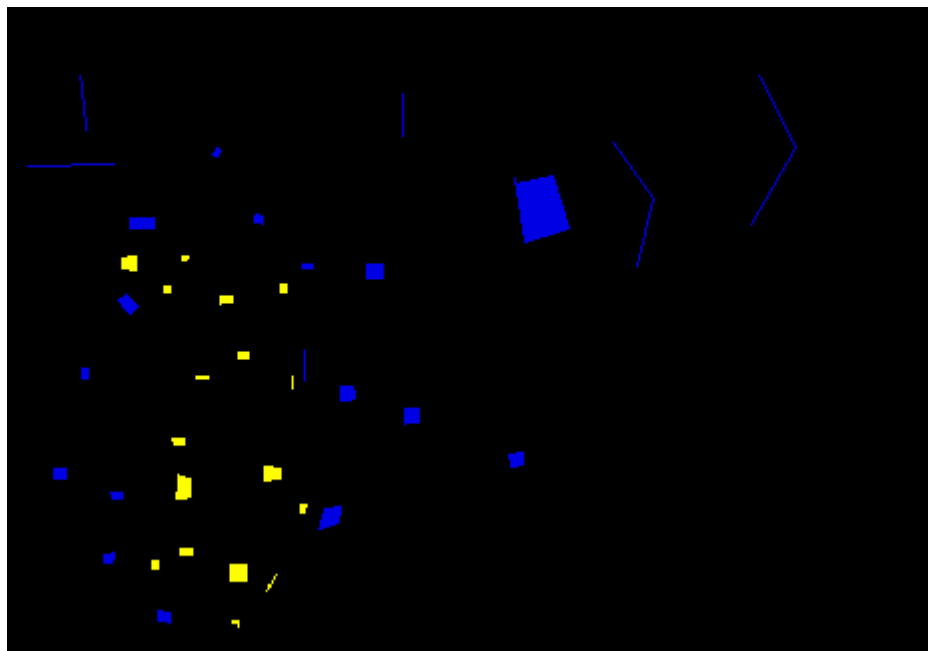
2004 junho



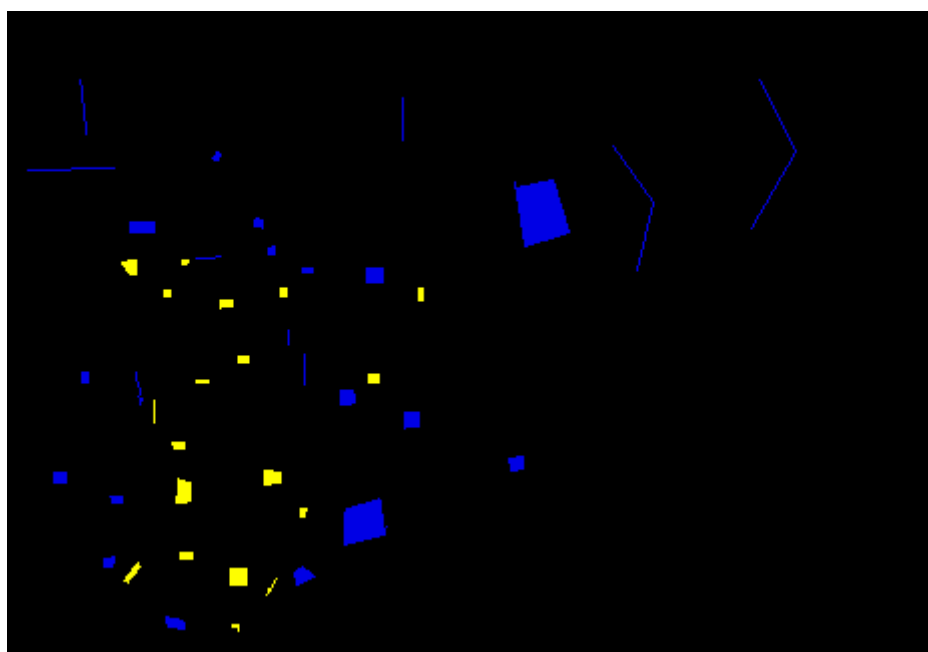
2004 julho



2005 Setembro



2007 junho



## Secção B

```
%CAPITULO II
%notas: classe 1: cabedelo
%      classe 0 e 2: não cabedelo

% imagem com classes (PCI).
diretorio_train=['D:\tese_mestrado\2_area\PCI\areas de
treino_teste\'];
train = ['2001D_train.TIF'; '2004JN_train.TIF'; '2004JL_train.TIF';
        '2005JN_train.TIF'; '2005S_train.TIF'; '2007JN_train.TIF'];

    for i=1:6 %nr de conjunto de dados (2001, 2004, 2004, 2005, 2005,
2007)
% i
%% leitura de PCT das imagens de treino
% A=[diretorio_train,train(i,:)]
    imagem_treino = imread([diretorio_train,train(i,:)]);

%tabulate(imagem_treino(:))
%%
% Leitura das 4 bandas.
    diretorio_img=['D:\tese_mestrado\2_area\imagens
editadas_8bt\2001_dec\';
        'D:\tese_mestrado\2_area\imagens editadas_8bt\2004_jun\';
        'D:\tese_mestrado\2_area\imagens editadas_8bt\2004_jul\';
        'D:\tese_mestrado\2_area\imagens editadas_8bt\2005_jun\';
        'D:\tese_mestrado\2_area\imagens editadas_8bt\2005_sep\';
        'D:\tese_mestrado\2_area\imagens editadas_8bt\2007_jun\'];

    red = imread([diretorio_img(i,:), 'red.tif']);
    green = imread([diretorio_img(i,:), 'grn.tif']);
    blue = imread([diretorio_img(i,:), 'blue.tif']);
    nir = imread([diretorio_img(i,:), 'nir.tif']);

%numero de classes
    nr_classes = 2;
% percentagem de areas de treino para aprendizagem
    prop_treino = 0.7;

for k = 1:nr_classes

    idx_aux = find(imagem_treino == k);
    red_aux = red(idx_aux);
    green_aux = green(idx_aux);
    blue_aux = blue(idx_aux);
    nir_aux = nir(idx_aux);

    %definicao dos indices de treino e teste (não é aleatório!!!)
    idx_rand = randperm(length(idx_aux));
    idx_treino = idx_rand(1:round(prop_treino*length(idx_aux)));
    idx_teste = idx_rand(round(prop_treino*length(idx_aux))+1:end);
%     S1=size(idx_treino);
%     S2=size(idx_teste);
    eval(['reflect_C', num2str(k), '_treino = [red_aux(idx_treino)
green_aux(idx_treino) blue_aux(idx_treino) nir_aux(idx_treino)];'])
```

# Árvores de Decisão Aplicadas à Detecção de Formas Costeiras Através de Imagens IKONOS-2

## Anexos

```
eval(['reflect_C', num2str(k), '_teste = [red_aux(idx_teste)
green_aux(idx_teste) blue_aux(idx_teste) nir_aux(idx_teste)];'])

end

%% definição dos anos das imagens
ano=['2001DC'; '2004JN'; '2004JL'; '2005JN'; '2005ST'; '2007JN'];

%% areas de treino para aprendizagem
reflect_todas_treino = [reflect_C1_treino
ones(size(reflect_C1_treino,1),1)
reflect_C2_treino ones(size(reflect_C2_treino,1),1)*2];

%exportar
% xlswrite(['dados_isprs_treinoteste',ano(i,:),'.xls'],
reflect_todas_treino, 'treino')

file_ID_tr=fopen(['C:\Users\dario\Documents\dados_isprs_treino',ano(i,
:),'.txt'], 'w');
fprintf(file_ID_tr,'%d\t %d\t %d\t %d\t %d\n', reflect_todas_treino);
fclose(file_ID_tr);

%% areas de treino para validação
reflect_todas_teste = [reflect_C1_teste
ones(size(reflect_C1_teste,1),1)
reflect_C2_teste
ones(size(reflect_C2_teste,1),1)*2];

% exportar
% xlswrite(['dados_isprs_treinoteste',ano(i,:),'.xls'],
reflect_todas_teste, 'teste')
file_ID_ts=fopen(['C:\Users\dario\Documents\dados_isprs_teste',ano(i,
:),'.txt'], 'w');
fprintf(file_ID_ts,'%d\t %d\t %d\t %d\t %d\n', reflect_todas_teste);
fclose(file_ID_ts);

%% numero das áreas de treino para cada classe
Classe1_tr=(size(reflect_C1_treino));
Classe1_ts=(size(reflect_C1_teste));
Classe2_tr=(size(reflect_C2_treino));
Classe2_ts=(size(reflect_C2_teste));
nr_treino(i,1:2)=[Classe1_tr(1) Classe2_tr(1)];

end

%% exportar numero das areas de treino para cada classe

file_ID=fopen('C:\Users\dario\Documents\numero_treino.txt', 'w');
fprintf(file_ID,'%d\t %d\n', nr_treino);
fclose(file_ID);

%% exportar imagens em colunas para R
nome_pasta = 'D:\tese_mestrado\2_area\imagens editadas_8bt\';
ano=['2001_dec'; '2004_jun'; '2004_jul'; '2005_jun'; '2005_sep';
'2007_jun'];

for i= 1:6
% Ler imagens do Cabedelo (4bandas de 6 anos)
```

## Árvores de Decisão Aplicadas à Detecção de Formas Costeiras Através de Imagens IKONOS-2

### Anexos

```
nome_fx_B1 = [nome_pasta, ano(i,:), '\blue.tif'];
nome_fx_B2 = [nome_pasta, ano(i,:), '\grn.tif'];
nome_fx_B3 = [nome_pasta, ano(i,:), '\red.tif'];
nome_fx_B4 = [nome_pasta, ano(i,:), '\nir.tif'];
B1=imread(nome_fx_B1);
B2=imread(nome_fx_B2);
B3=uint8(imread(nome_fx_B3));
B4=uint8(imread(nome_fx_B4));

%% eliminar duas linhas e uma coluna da banda grn
if i == 1
    B2_set=im2double(B2);
    B2_set(325:326,:)=[];
    B2_set(:,465)=[];
    B2=uint8(255*B2_set);
end

%% ler imagens em colunas
B_all_coluna = [B3(:) B2(:) B1(:) B4(:)];

%gravar vetor com os DN das 4 bandas
nome_out = ['C:\Users\dario\Documents\B_all_coluna_',ano(i,:), '.txt'];
file_ID=fopen(nome_out, 'w');
fprintf(file_ID, '%d %d %d %d\n', B_all_coluna);
fclose(file_ID);

end
```



Secção C

##CAP2 - Areas

```
rm(list=ls())
```

```
## aprendizagem
```

```
# numeros de areas de treino
```

```
nr_treino=read.table("numero_treino.txt", header=FALSE, sep="\t")
```

```
# anos das imagens
```

```
ano=c("2001DC", "2004JN", "2004JL", "2005JN", "2005ST", "2007JN")
```

```
# anos das imagens com diferente designação
```

```
ano2=c("2001_dec", "2004_jun", "2004_jul", "2005_jun", "2005_sep", "2007_jun")
```

```
#variáveis para armazenar as precisões
```

```
pKappa=c(seq(6)) #variavel para armazenar o parmetro estatistico kappa
```

```
pGlobal=c(seq(6)) #variavel para armazenar a precisão global
```

```
pUtilizador=matrix(0,6,2) #variavel para armazaenar a precisao do utilizador
```

```
pProdutor=matrix(0,6,2) # variavel para armazenar a precisao do produtor
```

```
#ciclo para 6 anos de imagens do cabedelo
```

```
for (k in seq(6))
```

```
{
```

```
  print(k)
```

```
  # Dados de treino
```

```
  dados_treino
```

```
  read.table(paste("dados_isprs_treino",ano[k],".txt",sep=""),sep="\t")
```

```
  dim_dados_treino=dim(dados_treino)
```

=

```
# dados de teste

dados_teste = read.table(paste("dados_isprs_teste",ano[k],".txt",sep=""),
sep="\t")

dim_dados_teste=dim(dados_teste)

# arvore de decisao com rpart

library(rpart)

dados=data.frame(dados_treino[,1:4] , classes = factor(c(rep("1",nr_treino[k,1]),
rep("2", nr_treino[k,2]))))

arvore=rpart(classes~., data=dados, method="class",cp=0,xval=0)

#ilustrar a arvore

dev.new(k)

plot(arvore,uniform=TRUE, main=paste("Árvore ",ano[k],sep=""))

text(arvore,digits=0,fancy=TRUE,font=6,fheight=0.8,cex=.8)

nome_out=paste("arvore",ano[k],".jpeg",sep="")

dev.copy(jpeg,nome_out)

#arvore de decisao com rpart (Poda, com cp=0.01 e numero de cv=5)

dados=data.frame(dados_treino[,1:4] , classes = factor(c(rep("1",nr_treino[k,1]),
rep("2", nr_treino[k,2]))))

arvore_prune=rpart(classes~., data=dados, method="class",cp=0.01,xval=5)

#ilustrar a arvore Podada

#dev.new(2+k)

#plot(arvore_prune,uniform=TRUE,          main=paste("Árvore          Podada
",ano[k],sep=""))

#text(arvore_prune,digits=0,fancy=TRUE,font=6,fheight=0.8,cex=.8)

#nome_out=paste("arvore_prune",ano[k],".jpeg",sep="")

#dev.copy(jpeg,nome_out)
```

```
#resumo dos erros das divisões

CP=printcp(arvore_prune)

write.table(CP,          paste("CP_",ano[k],".txt",sep=""),sep="\t",quote=FALSE,
row.names=FALSE, col.names=FALSE)


#ilustar os erros

#dev.new(3+k)

#plotcp(arvore_prune, main=ano[k])

#nome_out=paste("erro_relativo",ano[k],".jpeg",sep="")

#dev.copy(jpeg,nome_out)

#dev.off()


#sumario da arvore

sumario=summary(arvore)

#write.table(sumario,
paste("sumario_tree_",ano[k],".txt",sep=""),sep="\t",quote=FALSE,
row.names=FALSE, col.names=FALSE)


# matriz de perdas

L=residuals(arvore_prune)

write.table(L,
paste("matrix_perdas_",ano[k],".txt",sep=""),sep="\t",quote=FALSE,
row.names=FALSE, col.names=FALSE)


# exportar árvore

save(arvore_prune, file=paste("arvore_R",ano[k],".RData", sep=""))

save(arvore_prune, file=paste("arvore_R",ano[k],".txt",sep=""), ascii=TRUE)


## teste

# previsao

previsao = predict(arvore_prune, dados_teste[,1:4], type="class")
```

##precisão

```
# matriz de confusão

mc=table(dados_teste[,5], previsao) #matriz de confusão

write.table(mc,
paste("matrx_confusao",ano[k],".txt",sep=""),sep="\t",quote=FALSE,
row.names=FALSE, col.names=FALSE)


#pametros da avaliação da precisão
pG=0 #dados para iteração
pK=0 #dados para iteração
Tmc=sum(mc)      #nr total de pixeis
lim=dim(mc) #dimensao da matrix de confusao

for (i in seq(lim[1]))
{
  pU=0
  pP=0
  for (y in seq(lim[1]))
  {
    pU= pU+mc[i,y]
    pP= pP+mc[y,i]
  }
  pUtilizador[k,i]=(mc[i,i]/pU)*100 #precisao do utilizador
  pProdutor[k,i]=(mc[i,i]/pP)*100   #precisao do produtor
  pG=pG+mc[i,i]
  pK=pK+(pU/Tmc)*(pP/Tmc)
}

pKappa[k]=((pG/Tmc)-pK)/(1-pK) #parametro estatistico kappa
```

```
pGlobal[k]=pG/sum(mc)*100 #precisão Global
```

```
##leitura e previsão da imagem original
```

```
documento=paste("B_all_coluna_",ano2[k],".txt", sep="")
Linha_p_10=read.table(documento, header=FALSE, sep="")
img_prev=predict(arvore_prune,Linha_p_10,type=c("vector"))
write.table(img_prev, paste("previsao_",ano2[k],".txt",
sep=""),sep="\t",quote=FALSE, row.names=FALSE, col.names=FALSE)

}
```

```
#exportar dados da precisão
```

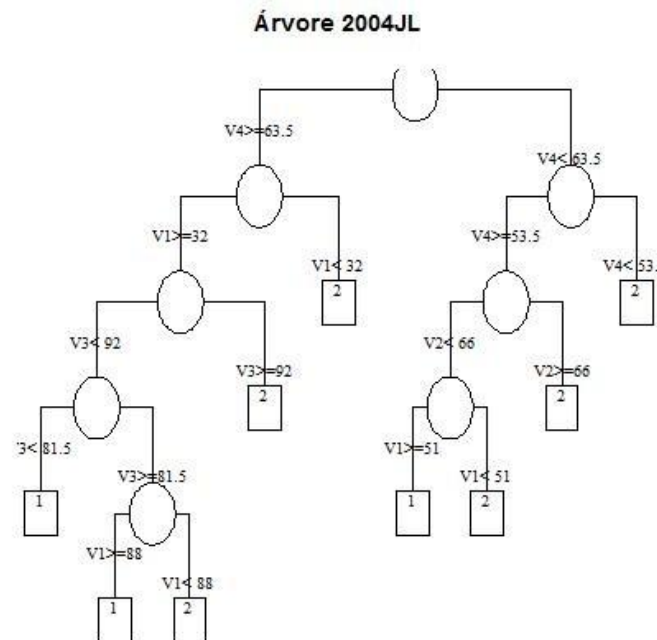
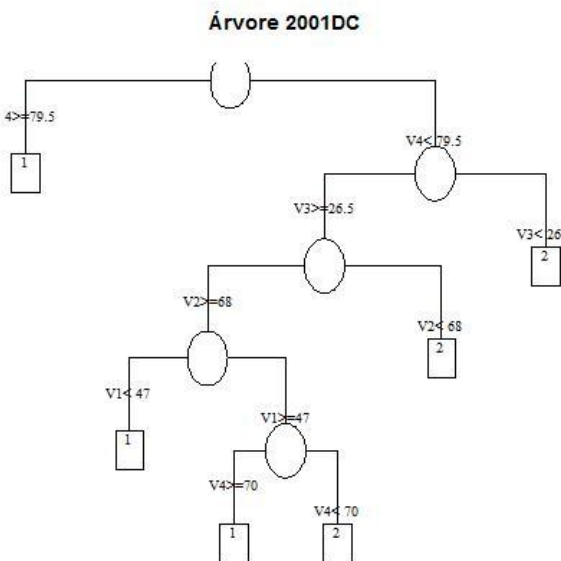
```
write.table(pKappa, "kappa_6anos.txt" ,sep="\t",quote=FALSE, row.names=FALSE,
col.names=FALSE)
```

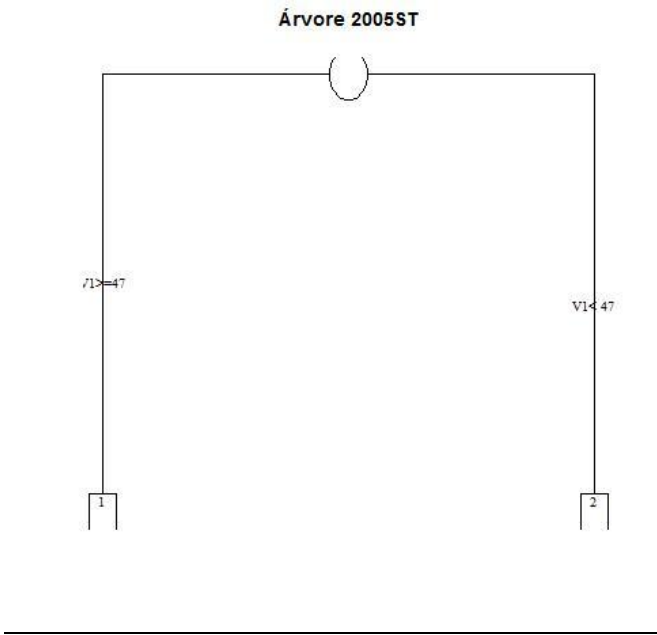
```
write.table(pGlobal, "OA_6anos.txt" ,sep="\t",quote=FALSE, row.names=FALSE,
col.names=FALSE)
```

```
write.table(pUtilizador, "precisao_utilizador_6anos.txt" ,sep="\t",quote=FALSE,
row.names=FALSE, col.names=FALSE)
```

```
write.table(pProdutor, "precisao_produto_6anos.txt" ,sep="\t",quote=FALSE,
row.names=FALSE, col.names=FALSE)
```

Secção D





### Secção E

```
%% apresentar as imagens
clc
clear all
close all

%% directorio das imagens [0 255] do cabedelo
directorio= 'D:\tese_mestrado\2_area\imagens editadas_8bt\';

% Ilustração das imagens classificadas do cabedelo
nova_pasta='C:\Users\dario\Documents\';
ano=['2001_dec';'2004_jun';'2004_jul';'2005_jun';'2005_sep';'2007_jun'
];

    for i=1:6

%% Ler imagens do Cabedelo (4bandas de 6 anos)
nome_fx_B1 = [directorio, ano(i,:),'\blue.tif'];
nome_fx_B2 = [directorio, ano(i,:),'\grn.tif'];
nome_fx_B3 = [directorio, ano(i,:),'\red.tif'];
nome_fx_B4 = [directorio, ano(i,:),'\nir.tif'];
B1=imread(nome_fx_B1);
B2=imread(nome_fx_B2);
B3=uint8(imread(nome_fx_B3));
B4=uint8(imread(nome_fx_B4));
if j == 1
    B2_set=im2double(B2);
    B2_set(325:326,:)=[];
    B2_set(:,465)=[];
    B2=uint8(255*B2_set);
end

%% leitura dos vetores de previsao
nome=[nova_pasta,'previsao_',ano(i,:),'.txt'];
P= textread(nome);

%mudar escala para (0,1)
ind2=find(P==2);
P(ind2)=0;

%dimensão das imagens
[Dim_img_lnh Dim_img_cln]=size(B3);

%redefinir o vetor em matrix
P_matrix=reshape(P',Dim_img_lnh,Dim_img_cln); %transforma o vetor em
matrix

%% Representação da imagem
X_aux1 = double(P_matrix);
X_aux2 = uint8((X_aux1-min(X_aux1(:)))/(max(X_aux1(:))-
min(X_aux1(:)))*255);
[X,map]=gray2ind(X_aux2,2);

mapa_cores = [0 0 0 % não cabedelo
255 255 255]; % cabedelo
mapa_cores = mapa_cores./255;
```



```
%area antes do processamento
ind_area_pre=find(X==1);
nr_pixeis=size(ind_area_pre);
area_pre(i)=nr_pixeis(1)*16; %16=resolução da imagem (4x4m)

% figure
% imshow(ind2rgb(X, mapa_cores), 'border', 'tight');

%% Pos-processamento
%% Eliminar zonas da água classificadas como cabedelo
%encontrar pixeis conectados
cc=bwconncomp(X);

%extrair pixeis conectados num area
stats=regionprops(cc, 'Area');

% indice dos pixeis conectados por menos de YYYYYY pixeis
ind800=find([stats.Area]>800);
X800= ismember (labelmatrix(cc),ind800);

%inverter as cores
ind_branco=find(X800==1);
ind_preto=find(X800==0);
X800(ind_branco)=0;
X800(ind_preto)=1;

%% preencher o cabedelo
%encontrar pixeis conectados
cc=bwconncomp(X800);

%extrair pixeis conectados num area
stats=regionprops(cc, 'Area');

% indice dos pixeis conectados por menos de YYYYYY pixeis
ind700=find([stats.Area]>700);
X700= ismember (labelmatrix(cc),ind700);

% inverter as cores
ind_branco=find(X700==1);
ind_preto=find(X700==0);
X700(ind_branco)=0;
X700(ind_preto)=1;

%área do cabedelo (m2)
ind_area=find(X700==1);
nr_pixeis=size(ind_area);
area(i)=nr_pixeis(1)*16; %16=resolução da imagem (4x4m)

figure
imshow(ind2rgb(X700, mapa_cores), 'border', 'tight')

end

%gravar areas do cabedelo
```

## Árvores de Decisão Aplicadas à Detecção de Formas Costeiras Através de Imagens IKONOS-2

### Anexos

```
nome_area_pre = ['C:\\Users\\dario\\Documents\\area do  
cabedelo_preprocessamento.txt'];  
file_ID=fopen(nome_area_pre, 'w');  
fprintf(file_ID, '%d\\t %d\\t %d\\t %d\\t %d\\t %d\\n', area_pre);  
fclose(file_ID);  
  
nome_area= ['C:\\Users\\dario\\Documents\\area do  
cabedelo_processamento.txt'];  
file_ID=fopen(nome_area, 'w');  
fprintf(file_ID, '%d\\t %d\\t %d\\t %d\\t %d\\t %d\\n', area);  
fclose(file_ID);
```